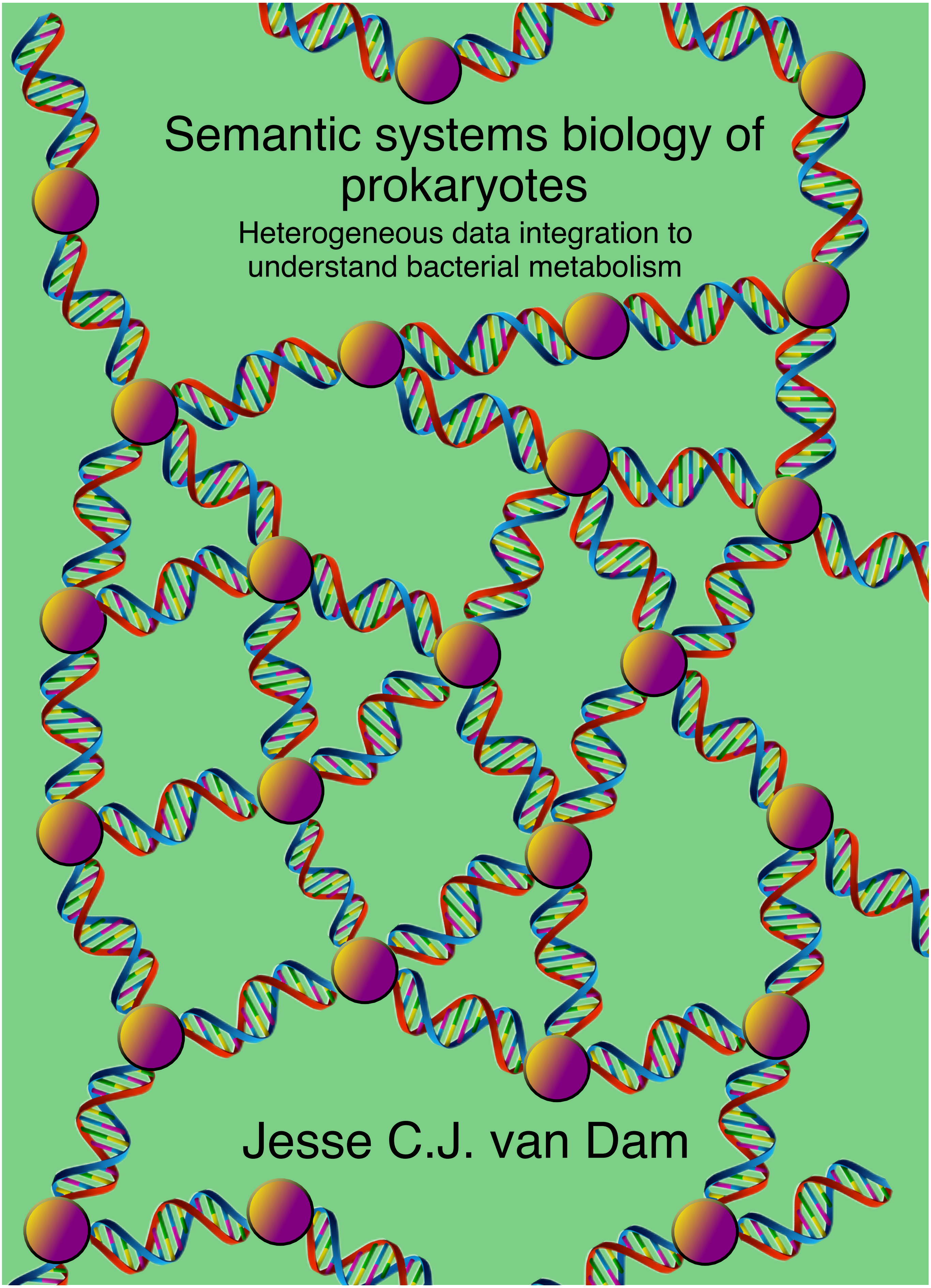


Semantic systems biology of prokaryotes

Heterogeneous data integration to understand bacterial metabolism

Jesse C.J. van Dam

Propositions

1. A term ontology is not a schema, therefore the GO ontology is not a schema.
   (this thesis)

2. Scientific data needs provenance for reproducibility.
   (this thesis)

3. In the life sciences, the dry lab cycle can improve wet lab cycles thereby reducing the cost per unit of knowledge gained.

4. The biggest gain in the field of data analyses can be achieved in the data preparation phase.

5. Repetition of scientific statements does not prove that something is true.

6. Semantic, web-based and decentralized data solutions that rely on continuous information flows will suddenly bring an end to the current social media platforms in the world.

7. The generation of data, creation of tools and extraction of knowledge needs different people, attitudes and infrastructures for optimal results.

8. To be green and sustainable we have to take control over evolution.

Propositions belonging to the thesis entitled

Semantic systems biology of prokaryotes

Jesse J.C. van Dam
Wageningen, 23th January 2019

Semantic systems biology of prokaryotes

Heterogeneous data integration to understand bacterial metabolism

Jesse C. J. van Dam

Thesis committee

Promotor
Prof. Dr Vitor A. P. Martins dos Santos
Professor of Systems and Synthetic Biology
Wageningen University & Research

Co-promotors
Dr María Suárez Diez
Assistant professor, Laboratory of Systems and Synthetic Biology
Wageningen University & Research

Dr Peter J. Schaap
Associate professor, Laboratory of Systems and Synthetic Biology
Wageningen University & Research

Other members
Dr Katy J. Wolstencroft, LIACS, Leiden University
Prof. Dr Jan L. Top, Wageningen University & Research
Prof. Dr Bedir Tekinerdogan, Wageningen University & Research
Prof. Dr Alexander Goesmann, Bielefeld University, Germany

This research was conducted under the auspices of the Graduate School VLAG (Advanced studies in Food Technology, Agrobiotechnology, Nutrition and Health Sciences).

Semantic systems biology of prokaryotes

Heterogeneous data integration to understand bacterial metabolism

Jesse C. J. van Dam

Thesis
submitted in fulfillment of the requirements for the degree of doctor
at Wageningen University
by the authority of the Rector Magnificus,
Prof. Dr A.P.J. Mol,
in the presence of the
Thesis Committee appointed by the Academic Board
to be defended in public
on Wednesday 23 January 2019
at 4 p.m. in the Aula.

Jesse C. J. van Dam
Semantic systems biology of prokaryotes,
255 pages.

PhD thesis, Wageningen University, Wageningen, the Netherlands (2019)
With references, with summary in English

ISBN: 978-94-6343-550-5
DOI: 10.18174/464871

Contents

Preface

In 2009 my brother got me interested in processing microarray data. We noticed that a lot of the expensive data was only used to answer the question of its creators, which we found a waste of resources, because we had the feeling that the data contained much more valuable information. A method to extract this information would be to combine all the data sets to find genes related to each other. So, we started to create large scale mutual information networks and we tried to make a useful network visualization to gain new insights. With a lot of enthusiasm and in my free time, I worked to push past the hairballs we got from the initial mutual information networks. In doing so, I came close to reinvent the CLR[161] and Aracne[48] methods. This process captured so much my interest that in 2010 I started my master in Bioinformatics, which allowed me to apply my knowledge from the field of information technology to biology.

In 2011 I started my master thesis with the objective of gaining knowledge from omics expression data. The project exploited data of more than 200 environmental and gene knockout perturbations of *Mycobacterium tuberculosis* (MTB). In my master thesis I started to further extend my network based approach. I started using this approach to integrate other types of information, such as gene neighborhood, gene co-occurrence, BLAST similarity obtained from the STRING website [246] and a metabolic pathway map obtained from Pathway Tools (PT) [264]. Once the large amount of available information was represented as networks, I started to focus on the data integration process and started developing a graphical user interface to compare networks.

In the spring of 2012 I started my internship in Leiden at LUMC in the group of Bio-Semantics, working on the nano publication system. There I got to know about data reuse and integration issues and about semantic web technologies and their potential to solve these issues.

In the summer of 2012 I started my PhD in the laboratory of Systems and Synthetic Biology at Wageningen UR. The goal was to apply these concepts to genome scale modeling of metabolism and its regulation. At the start of my PhD, I continued with the work of my master thesis and to integrate more resources and information, for instance by adding cluster data obtained from cMonkey [409]. However, I noticed that pre-generated data was not enough and started to directly integrate analysis methods. One of these was a method to find motifs within a selected set of genes, the motif was subsequently used to find new genes that could be integrated in the cluster. In the process, we noticed that we needed to include additional data, such as genome annotation in order to answer more complex biological questions. At this point, the process became too cumbersome and too time consuming to continue any further as it would have taken weeks of coding to answer one single biological question. For example, what are all known cofactors used to transfer electrons and which (set of) domains are specifically associated to one type of cofactor or type of substrate. Therefore, I started to develop solutions based on semantic web technologies that would help overcome this problem and would allow me continue on my conquest to reuse and integrate more and more data.

Within the field of biology I am particular interested in the apparent modularity and systematic setup of the metabolic and regulatory systems within prokaryotic organisms. To understand these systems more deeply one can use genome scale metabolic models. Therefor, I continued my efforts to reuse and integrate more and more omics data for the generation of genome scale metabolic models of prokaryotes. It is this interest which resulted in the work presented in this thesis.

Chapter 1

Introduction

Systems Biology advocates for a system-wide perspective in which we try to comprehend the functions of the components and their relationships in their cellular embedding within the living organism. With the advance of sequence technologies, the amount of omics data available in public data repositories is growing exponentially [450]. The exponential growing volume of available omics data empowers top-down approaches, combining system-wide data with modeling to gain insights into the molecular networks under study [84]

Increasing concern regarding climate change is prompting society to move from the oil-based economy [117]. Emergence of extensive resistant strains also requires the development of new intervention strategies [208]. Systems biology approaches are currently used to gain knowledge on microbial organisms with the goal of developing more efficient mechanisms to kill pathogenic bacteria or to develop new strains for microbial production. Systems biology approaches have also been used, among others, to identify mechanisms related to pathophysiology; select novel drug targets and biomarkers; assess patient risk; and to develop interventions aimed at recovering homeostasis within complex communities [109].

Top-down approaches entail (omics) data processing and modeling to turn it into actionable knowledge. This process can be described using the Data-Information-Knowledge-Wisdom (DIKW) pyramid model depicted in Figure 1.1. This model is used in the Information Sciences field [517]. The DIKW model describes four steps in the overall process of data value extraction: data collection, data to information conversion, from information to knowledge and from knowledge to wisdom. The review by Rowley [421] includes definitions from multiple authors for data, information, knowledge and wisdom. In the following I will provide definitions for these concepts that will be used through this thesis. It should be noted that still there is no firm agreement on these terms and differences could appear between the definitions here provided and those of other authors.

Data are unprocessed experimental measurements that are accessible in a digital system associated with a set of meta data. *Information* is processed and structured data that can be integrated, processed, queried and curated. *Knowledge* is the actionable set of information, that can be used to inform decisions needed for new applications, for the generation of new data, information and

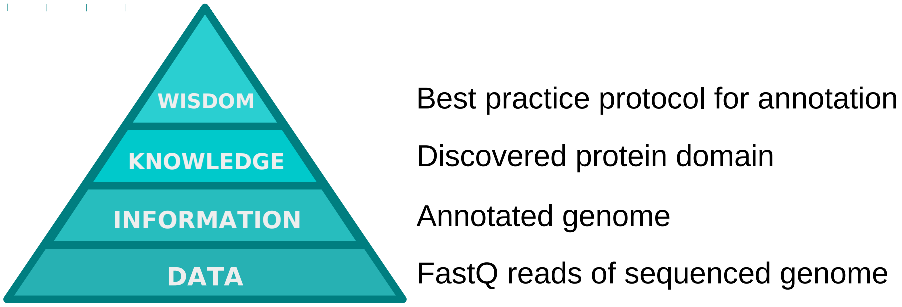

WISDOM
KNOWLEDGE
INFORMATION
DATA
Best practice protocol for annotation
Discovered protein domain
Annotated genome
FastQ reads of sequenced genome

FIGURE 1.1: **DIKW pyramid model** for knowledge management. An example pertaining biological data is given on the right. For example data can be FastQ reads of a sequenced genome; information can be an annotated genome; knowledge can be a discovered protein, with a functional description; and wisdom can be a best practice protocol for annotating a genome.

knowledge or for the development of tools that do so. Three classes of knowledge can be distinguished: *Human based knowledge*: knowledge stored in the brain. This includes the practical knowledge one gains when working on a given topic for a long time or after analyzing a new set of data and information. *Text based knowledge*: knowledge that is stored in any form of human language. *Computer based knowledge*: structured knowledge that can be used by computer based tools. Examples of this last type of knowledge are the UniProt database [50], the set of HMM models in InterPro [170] or the set of GEMs in the BiGG Models database [269]. Not all reviewed literature has a definition for wisdom, which remains the most elusive concept, here I will define *Wisdom* as the consensus subset of knowledge that is actively applied at the point of time. These definitions imply that the set the of knowledge is ever growing, whereas the set of wisdom is not. Knowledge that is no longer needed or actively being used will disappear in time from the wisdom, for example the knowledge on how to preprocess microarray data. Another important concept to define is *curation*: the process in which each element of information is assessed and accepted (or rejected) as valid knowledge or the process in which existing or new supporting or disproving information is attached to an existing element of knowledge.

These concepts will be used in this thesis to describe the extraction of value from biological data and their transformation into knowledge and wisdom within the framework of systems biology. In the general work flow that will be described, *data* is generated as the output of an experiment. In this experiment a biological sample is generated which gets sequenced or measured, resulting in a (omics) data set with an associated meta data file. The meta data file describes each sample and possibly each sequence or measurement. Next, a bio-informatics pipeline is applied to turn these data and meta data into information, for example, the assembly and annotation of a newly sequenced genome with associated phenotype such as Gram staining, optimal growth temperature and duplication time. Then, a researcher interprets this information, thereby creating new knowledge that can be applied to design new experiments or to further improve the data to information and information to knowledge processing. An example of a knowledge driven application is the genome editing tools facilitated by the knowledge on the CRISPR/Cas9 system [518]. Follow up experiments can be designed, for instance, to test hypothetical protein-protein interaction. Inclusion of newly identified Hidden Markov Models (HMM) the Pfam [169] database is an example of how the data to knowledge process is further improved. The last step in the DIYW model is achieved overtime as researchers work with the knowledge stored in databases and a subset of this knowledge becomes common good, which is the wisdom of that time. An example of wisdom would be protocols and standard operating procedures for genome sequencing, assembly and annotation. Finally, current knowledge and wisdom can be used to inform the design of new experiments, which result in the creation of new data. This last step closes a cycle, which is typically referred to as 'the wet-lab cycle'. However, knowledge and wisdom can be used to create new methods to transform data into information. This forms a shorter cycle which excludes the execution of new experiments, therefore we will call this cycle the dry-lab cycle and it is depicted in Figure 1.2

Big data challenge

Challenges have arise in the process of knowledge and wisdom extraction due to the large increase of available (omics) data sets. Thus, analytical methods have to be developed to extract better insights from these big data. An important distinction exist between *Horizontal* and *Vertical* data processing and integration, which is depicted in Figure 1.3.

The number of omics data set types is limited: genomics, transcriptomics, proteomics, metabolics and phenomics, but it is slowly growing as new high throughput measurements become available, such as lipidomics. However, the number and sizes of these omics data sets are constantly growing [450]. Moreover, with the use of high throughput techniques many combinations of conditions can be tested for example, environmental conditions, gene knockouts and sets of species. This results in a growth in diversity and size of the associated meta data [165]. As a result, the number of testable combinations for which enough statistical power can be achieved is rapidly growing. These, in turn leads to an increasing number of applicable methods, each producing unique sets of information. This information requires (manual) integration, processing and curation to turn it into knowledge. The great diversity of information results int a bottleneck in the complete process.

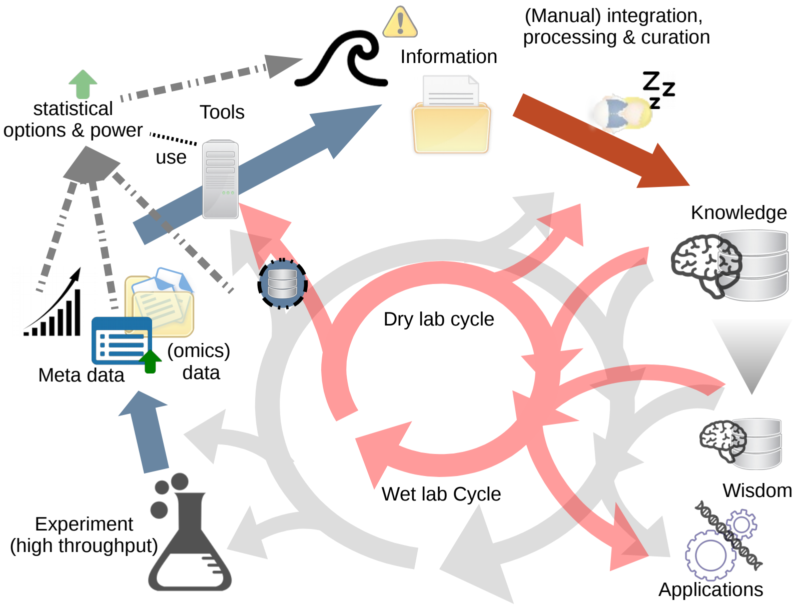


FIGURE 1.2: **Wet and dry-lab cycles in systems biology.** Experiments are designed based on currently available knowledge and wisdom. Execution of the experiments results in new data with associated meta data. Data and meta data are subsequently transformed into information through tools that use existing knowledge and information. The generated information is subsequently processed into knowledge through (manual) integration, processing or curation efforts. The knowledge can become part of the current wisdom, which can be used to create new applications and new experiments thereby closing the wet-lab cycle. The dry-lab cycle, however, excludes new experiments. Instead it uses (new) knowledge and wisdom to improve the tools and methods to generate information, knowledge and wisdom from data. In both cycles the step from information to knowledge is often the rate limiting step.

Semantic web technologies

Data heterogeneity often implies that answering a single biological question based on the ever growing collection of available information and knowledge might require days or even weeks of coding. Semantic web technologies can be used to (partly) overcome this problem [17].

Semantic web technologies have emerged as the result of the technological developments in computer sciences aimed at transforming the Internet from a network of linked documents into a semantic web of interlinked data which is meaningful to computers [58]. Early efforts to store knowledge for artificial

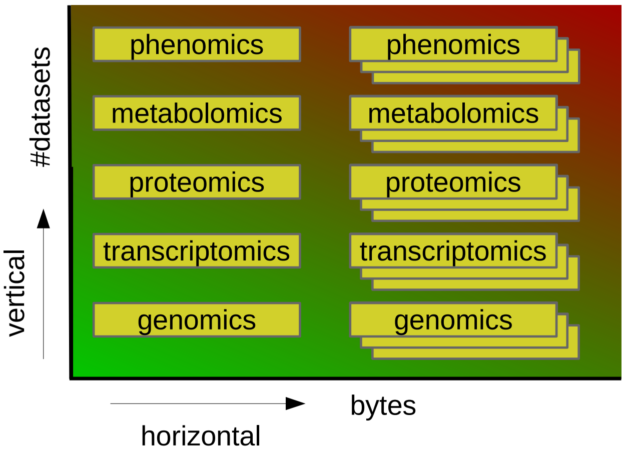


FIGURE 1.3: **Horizontal and vertical data integration scaling.** Horizontal refers to the size of the data sets in bytes and compute time needed to process them whereas vertical is related to the amount of heterogeneity within the data sets. The color indicates the relative difficulty of each process, where green indicates relatively easier tasks and red more difficult ones. The combination of performing both horizontal and vertical scaling is the most difficult problem to solve.

intelligence based systems resulted in the creation of General Frame Protocol [190], which is applied in the MetaCyc database [95]. This and other similar efforts were subsequently converted into the RDF standard [220]. This standard has become the default format for storing semantic web data. In RDF, knowledge is represented as set of triples forming a graph of linked data. Each RDF triple links a uniquely identified subject through a uniquely identified predicate to an object, which is either another subject or a literal, as depicted in Figure 1.4.

A subject represents an entity such as a person, a file or a gene whereas a literal is a value, which can either be a text, numerical value or a date among other. The unique identifier is an *International Resource Identifier* (IRI) which extends the *Uniform Resource Identifier* (URI) which is the well known unique identifier for a web page. In this way, the IRI "`http://example.com/cytidineDeaminase`" can be used to uniquely identify the subject representing the cytidine deaminase reaction. Interlinked triples, such as the ones in Table 1.1 can be used to generate a knowledge graph, as shown on Figure 1.5.

Using the RDF standard, a knowledge graph can be conceptualized into an RDF resource, that can be serialized into a N3, Turtle, or XML/RDF file. Semantic data can be loaded into a database, also referred to as triple store. If

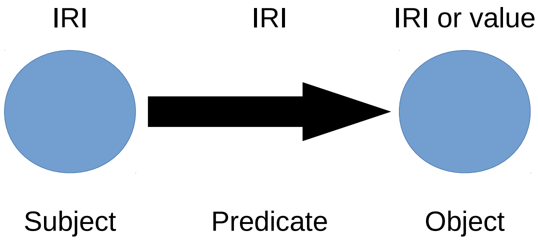


FIGURE 1.4: **RDF triple.** A triple consists of: a subject, a predicate and an object. The subject is identified with an IRI; The predicate defines the type of relationship which is identified with an IRI; The object, can be either another subject or a value. A value can be either a number, a string or a Boolean.

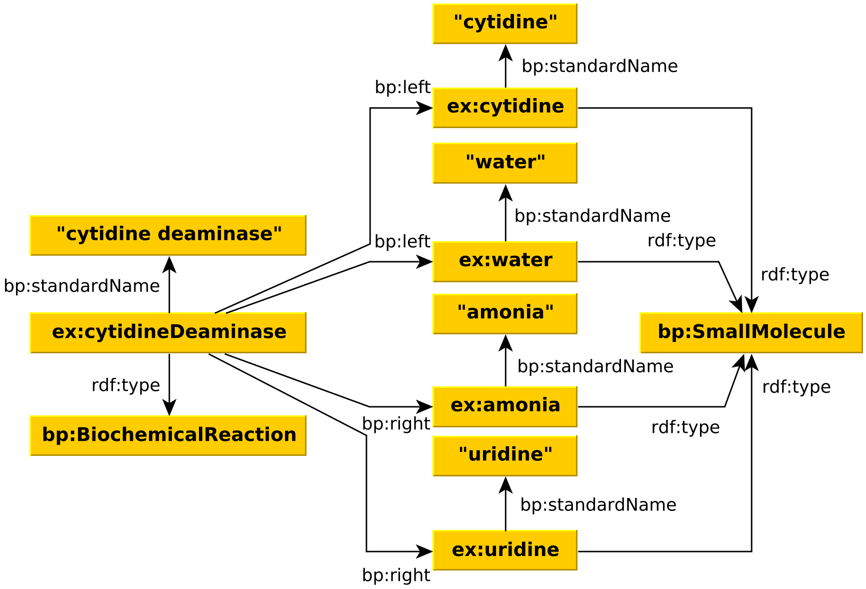


FIGURE 1.5: **RDF knowledge graph describing Cytidine deaminase reaction.** Description of the cytidine deaminase reaction based on interlinking the triples given in Table 1.1. Here the cytidine deaminase reaction is identified with "ex:cytidineDeaminase". Note that in this case the prefix "ex:" has be used to abbreviate the IRI "`http://example.com/`".

multiple RDF resources, each containing subjects sharing IRIs are loaded, the

TABLE 1.1: **Cytidine deaminase reaction RDF triple set**. These triples form the knowledge graph shown in Figure 1.5

Subject	Predicate	Object
ex:cytidineDeaminase	bp:standardName	"cytidine deaminase"
ex:cytidineDeaminase	rdf:type	bp:BiochemicalReaction
ex:cytidineDeaminase	bp:left	ex:cytidine
ex:cytidineDeaminase	bp:left	ex:water
ex:cytidineDeaminase	bp:right	ex:amonia
ex:cytidineDeaminase	bp:right	ex:uridine
ex:cytidine	rdf:type	bp:SmallMolecule
ex:cytidine	bp:standardName	"cytidine"
ex:water	rdf:type	bp:SmallMolecule
ex:water	bp:standardName	"water"
ex:amonia	rdf:type	bp:SmallMolecule
ex:amonia	bp:standardName	"amonia"
ex:uridine	rdf:type	bp:SmallMolecule
ex:uridine	bp:standardName	"uridine"

resources are automatically interlinked. Once the data is loaded, the query language SPARQL can be used to access, integrate and query the data. SPARQL is similar to SQL, but SPARQL operates on graphs instead of on tables. Moreover, SPARQL uses a graph pattern matching system that automatically performs the joins whereas joins have to be explicitly stated in SQL [185].

Knowledge graphs have no predefined structure nor a schema, however the structure is essential for reusable data and for consistent querying. Within conventional computer engineering, file structure is defined in a schema, for example the structure of an XML file is described in an associated DTD file. Within the semantic web world, an ontology is the equivalent of a schema. However, in my opinion this concept has a broader meaning. The book by Guarino et al. [209] includes a detailed definition for the concept ontology. For this thesis, I define *ontology* as a resource that supports the conceptualization and interpretation of another data or knowledge entity. I define *schema* as a document that defines the structure of a computer readable file. From these definitions, it follows that the ontology definition encapsulates the schema definition but not vice versa.

RDFS and OWL are two related standards to define an ontology. RDFS can be used to define the structure of an RDF resource. In this standard, each object can be defined as an instance of a class and each link as the realization of a property. This standard also allows subclasses to be defined. An example on how these concepts can be used to describe transcription regulation is provided in Figure 1.6.

OWL can be used to define concepts, which are commonly defined within a separate ontology. For instance, in Figure 1.6, the “GO:0006281: DNA repair

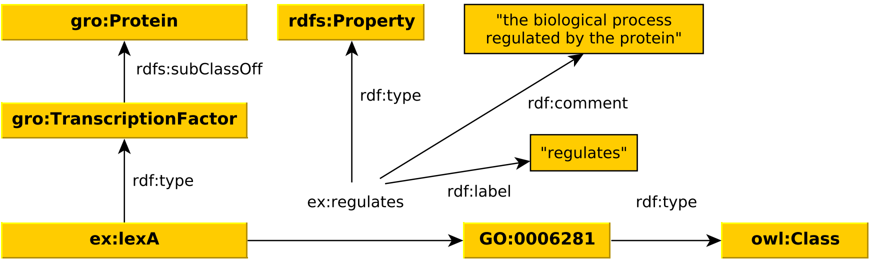


FIGURE 1.6: **LexA OWL example.** LexA is a transcription factor that regulates DNA repair process in prokaryotes [276]. LexA is an instance of the class transcription factor, which in turn is a sub class of the protein class. Every transcription factor is also a protein. The link regulates between the instance LexA and the concept(an owl:Class) DNA repair process (GO:0006281) is a realization of the property regulates. The property regulates is an RDFS property and has a label and comment.

process" concept is described in the GO ontology [92]. OWL extends RDFS and is a formal description of the conceptualization of the information in a knowledge graph. In this way, it defines the structure and relationships of the encoded knowledge [18]. OWL originated in the field of artificial intelligence [74] and is used to relate concepts through subclass-of relationships [423]. Using these standards, reasoners can be used to automatically find inconsistencies and recognize unknown or unclassified concepts [197].

Precise definition of the structure of a resource can also be used to validate the resource itself. If all subjects in a knowledge graph are defined as instances of some class or concept and all links are realizations of some property, then these definitions can be used to know which links are to be expected for each instance. This can be used to consistently query the resource. The class and property definitions themselves can also be stored within the same RDF resource and be used within the same query. However, RDFS and OWL have limited use to validate instances and associated link structure, because they are based on the open world assumption instead of the closed world assumption. In the open world assumption, everything the agent does not know is interpreted as undefined, while with the closed world assumption everything the agent does not know is interpreted as false [423]. If a class states that an associated property is obligatory, an instance not having the corresponding link would be wrong. However, in an open world assumption, the reasoner interprets this as: "I do not know if it is existing somewhere in world, therefore I reason this to be correct.

Therefore, to validate the structure of a knowledge graph, two new standards ShEx and SHACL have been developed based on the closed world assumption. These standards can be used on top of RDFS and OWL. Using ShEx or SHACL, each instance can be mapped to a shape definition. The shape definition includes the presence, multiplicity, and types and values of the links so that the links and the linked nodes can be validated. These validation standards can be used to ensure that the information stored in an RDF resource adheres to the predefined structure. This ensures the consistent storage of the relevant information.

SPARQL vs SQL and other table based solutions

Semantic web technologies are suited for data integration better than SQL and other table based solutions. Within SQL, data integration is based upon table coupling, which can only be done trough the use of primary and foreign keys pairs [51]. This means that to integrate two sets of databases, the table has to be redesigned. Furthermore, to retrieve and store data from and into an SQL table normalization or de-normalization steps are required. The data usage or generation from or into an RDF graph is more direct. The normalization and de-normalization steps typically involve coding work, that is particular expensive in scientific research. The amount of coding work can be reduced through semantic web based solution. Although SPARQL and SQL have many similarities, SPARQL is, in my opinion easier to use. The main reason is that SQL requires manual definition of the joins, whereas in SPARQL this is done automatically. Within the years that I have worked with SPARQL, I experienced that even non technical users where able to modify and create their own queries in a relatively short amount on time. On the other hand SQL databases are commonly faster than semantic web technology based databases. One reason, is that SQL is a more mature technology. But the fundamental reason is that when one performs a SPARQL query it requires all data to be joined, whereas in SQL one can create pre-joined tables.

Genome scale metabolic models (GEM)

Whole bacterial genome sequencing has become common practice [284]. This has resulted in completely sequenced genomes that can be used to predict phenotypes such as pathogenicity [296], physiological properties [361, 420], antibiotic resistance [275] and metabolic phenotypes. Metabolic phenotypes include auxotrophies and media requirements, anabolic and catabolic potential and biomass composition. These metabolic phenotypes can be predicted with Genome Scale Metabolic Models (GEM).

A GEM includes the complete set of biochemical reactions related to the species of interest. The GEM is represented as a network of nodes, in which each node is either a metabolite or a reaction. Simulations using constraint based GEM are performed under the steady state hypotheses, that states that none of the intracellular metabolites accumulates nor depletes. This hypothesis enables to calculate for each reaction possible flux values compatible with the steady state assumption. Additionally, one reaction is included in GEMs which represent the production of biomass. This reaction consumes all metabolites needed to produce one unit of biomass, which includes amino acids, nucleotides, vitamins, lipids and glycogens. Flux balance analysis is a commonly used technique to determine how an organism modifies its fluxes so that biomass production is maximized, for a given media. The media is represented as a set of 'influx' reactions that produce each of the metabolites within the simulated medium, see Figure 1.7.

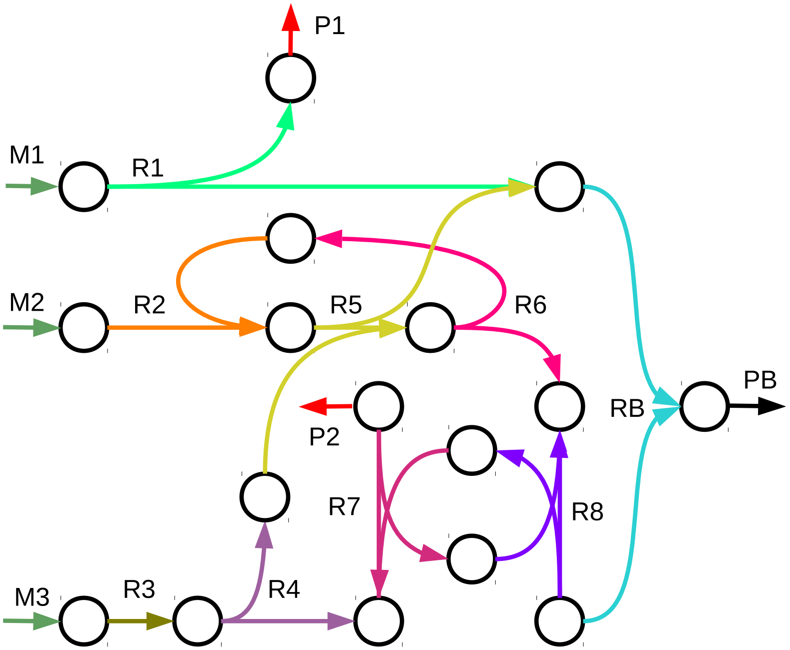


FIGURE 1.7: **Example metabolic network.** R1 to R8 represent the metabolic reactions within the organism. M1 to M3 simulate the media compounds consumed by the organism. Fluxes through these reactions are limited to the maximum uptake rates known for the organism. P1 and P2 represent production of metabolites or side products, for example CO_2. Finally, PB represents the biomass reaction and contains the list and stoichiometry of metabolites needed to produce one unit of biomass, this includes amino acids, nucleic acids, fatty acids, vitamins and cell wall components. Flux trough PB is maximized to simulate maximum growth given the measured uptake rates.

GEMs have a broad range of applications. A GEM can be used to predict auxotrophies thereby informing media design [25]. A GEM can also be used to find possible intervention options to kill pathogenic bacteria within a host, without killing the host cells and thus helps in the search for new therapeutic strategies and new antibiotics [221]. Finally, GEMs can also be used to design mutant strains for production of target compounds [221].

GEMs are built based on functional genome annotations. Functional annotations are used to map to reaction databases to identify reactions the organisms can perform. It is essential that all reactions use the same identifier for the compounds consumed and produced, otherwise the GEM will be inconsistent. Many metabolites can be identified with multiple identifiers, therefore standard name spaces like KEGG [260], MetaCyc [95] and the Model Seed [137] have a unique identifier for each compound. Still, it is difficult to to translate from one namespace to another, so special, cross linking databases like MetaNetX have been developed [343].

GEM building entails identifying, for each annotation, a reaction within the selected name space. However, the gene functional description does not always contains a direct link to the reaction within the selected name space. Therefor, automated GEM generation often results in gaps (missing reactions) that prevent the functioning of the GEM. This has lead to the development of gap-filling methods that iteratively modify the GEM until a functional GEM is achieved [545].

Within this protocol to create a working GEM, a lot of information needs to be integrated. Furthermore gap filling methods can be used to integrate additional phenotype data.

Outline

The goal of this thesis is to improve the genotype to phenotype associations with a focus on metabolic phenotypes of prokaryotes. To fulfill this goal I integrated data and for this task I developed supporting solutions using semantic web technologies.

To understand the phenotype and (pathogenic) systems of *M. tuberculosis* (Mtb), we created methods and tools to integrate and visualize available data which included genomics, transcriptomics, and proteomics data sets together with existing literature. The generated tools rely on synchronized network representations of each information source (such as correlated gene expression, gene neighborhood, gene co-occurrence, gene similarity, protein-protein interaction among others). The network visualization is embedded within a Data integration, visualization and analysis (DIVA) framework to manually compare and integrate the knowledge captured within these networks. The tool allows to generate pipelines to analyse other data types, such as ChIP-seq data. A prototype of this tool is presented in chapter 2. The prototype was initially used to investigate gene regulation in Mtb, specifically to analyse regulation of processes related to survival of the pathogen during infection, such as dormancy, zinc uptake and response to DNA damage. The stand alone prototype was later adapted as a Cytoscape application plugin, presented in chapter 3. The DIVA tool was also extensively used to analyze virulence strategies of this pathogen and an extensive review is provided in chapter 4.

The network based approach is limited as (functional) genome annotation can not be represented as a network. A different strategy had to be developed to better associate genotypes and phenotypes. Moreover, genome annotation is essential to develop GEM which provide the means to better understand and predict metabolic phenotypes. Therefore, new methods based on semantic web technologies were developed to retrieve and store genome functional annotations.

UniProt RDF contains gene annotation in the semantic web format, however accessing resources in this format is challenging. It was often found that the documentation was incomplete and the schema definition did match match to actual data structure within the resources. This makes it difficult to query the resource, because one can not know which information is available and one does not know which structure the data has. To overcome this issue, in chapter 5 we present RDF2Graph a tool that, can automatically extract structure information from an RDF resource.

Another challenge was associated to the used ontologies, as none of the existing ontologies (such as SO [154], FALDO [68] and SBOL [182]) fulfilled all the requirement needed for complete genome annotation. A new ontology for storing genome annotations was developed and the GBOL ontology together with a set of supporting tools was is presented in chapter 6. However, the ontology by itself does not produce usable data. We developed a conversion module to convert GenBank data into the GBOL format. However, for comparative genomics it is important that all annotations are created with the same methods and the contextual provenance, which captures the e-value scores, is present. Therefore, we created SAPP, an annotation pipeline that is presented in chapter 7. SAPP uses existing annotation tools, but wraps the results with associated provenance into GBOL ontology.

During the development of SAPP and GBOL, we noticed that it was hard to generate the correct RDF output. We encountered errors such as typing errors in the predicates, instances with missing attributes, instances that did have a non-unique IRI, and instances that had no type defined, among others. Furthermore we noticed that the time needed to consistently encode the more complex data structures was becoming a limiting factor. To overcome these problems Empusa was invented as presented chapter 8.

Subsequently, we used Empusa to improve and extend GBOL and SAPP, which we subsequently successfully used as an enabler to integrate annotation data and perform advanced comparative genomics. Specifically, we compared 432 Pseudomonas strains through integration of genomic, functional, metabolic and expression data, as presented in chapter 9

GEM are useful tools to perform predictions on metabolic phenotypes. Therefore, a pipeline to build complete GEM is presented in chapter 10. The pipeline includes a gap filling method that specifically integrates phenotype characterizations that are measured with high throughput methods. The method was extensively tested and benchmarked against existing approaches, as presented in chapter 10.

Finally, in chapter 11 I will discuss how the methods, tools and analysis presented in these thesis contribute to make better phenotype to genotype associations and I will discuss the opportunities associated to semantic web technologies in the life sciences, on their potential to expand the dry-lab cycle and on the data requirements for this task.

Chapter 2

Integration of heterogeneous molecular networks to unravel gene-regulation in *Mycobacterium tuberculosis*

Adapted from:

Jesse CJ van Dam, Peter J Schaap, Vitor AP Martins dos Santos, and Maria Suarez-Diez. "Integration of heterogeneous molecular networks to unravel gene-regulation in *Mycobacterium tuberculosis*". In: *BMC Systems Biology* 8:111 2014.

Abstract

Background: Different methods have been developed to infer regulatory networks from heterogeneous omics datasets and to construct co-expression networks. Each algorithm produces different networks and efforts have been devoted to automatically integrate them into consensus sets. However each separate set has an intrinsic value that is diluted and partly lost when building a consensus network. Here we present a methodology to generate co-expression networks and, instead of a consensus network, we propose an integration framework where the different networks are kept and analysed with additional tools to efficiently combine the information extracted from each network.

Results: We developed a workflow to efficiently analyse information generated by different inference and prediction methods. Our methodology relies on providing the user the means to simultaneously visualise and analyse the coexisting networks generated by different algorithms, heterogeneous datasets, and a suite of analysis tools. As a show case, we have analysed the gene co-expression networks of *M. tuberculosis* generated using over 600 expression experiments. Regarding DNA damage repair, we identified SigC as a key control element, 12 new targets for LexA, an updated LexA binding motif, and a potential mismatch repair system. We expanded the DevR regulon with 27 genes while identifying 9 targets wrongly assigned to this regulon. We discovered 10 new genes linked to zinc uptake and a new regulatory mechanism for ZuR. The use of co-expression networks to perform system level analysis allows the development of custom made methodologies. As show cases we implemented a pipeline to integrate ChIP-seq data and another method to uncover multiple regulatory layers.

Conclusion: Our workflow is based on representing the multiple types of information as network representations and presenting these networks in a synchronous framework that allows their simultaneous visualization while keeping specific associations from the different networks. By simultaneously exploring these networks and metadata, we gained insights into regulatory mechanisms in *M. tuberculosis* that could not be obtained through the separate analysis of each data type.

Background

Current biology research generates an ever-increasing deluge of omics derived data. Each type of omics data pertains to a single level of the biological system under investigation (transcriptomics, proteomics, metabolomics, lipidomics, etc.). While detailed knowledge of the individual genes, transcripts, proteins, metabolites and other cellular components remains important, understanding a biological system requires considering the networks into which these components are embedded and of their functioning as a (dynamic) whole. A major challenge in Systems Biology lies thus on developing effective and efficient methods to optimally extract the information contained in the aggregate of these datasets.

The increasing availability of genome-scale expression data has boosted the development of methods to infer the underlying regulatory networks. A broad range of alternative methods are available, see [40, 130, 512] for reviews, and each of them uses different mathematical tools and/or biological assumptions. A class of methods use differential equations to express transcript changes as a function of the transcript levels of the corresponding transcription factors [70, 91]. A second class of methods rely on Bayesian networks to analyse the joint probability distributions obtained from the experimental data [178, 536]. Other methods use the similarity between gene expression profiles to detect associations and to reconstruct a genome scale transcriptional regulatory network [48, 161, 410]. Another class of methods use a combination of machine learning techniques to produce prioritized lists of transcription factors regulating each target gene [234, 457]. Each method has different strengths and weaknesses, even methods using similar conceptual tools. For example C3NET [13] uses mutual information (MI) to reconstruct the core regulatory interactions in the network, that is, to recover the strongest interactions. Within this core, C3NET was shown to outperform other methods also using MI such as CLR [161] and ARANCE [48]. Each method provides different results, therefore much effort has been devoted to generate consensus networks from the multiple solutions. It has been shown that in many instances, an integrative approach combining the outcome of each algorithm produces the best result [477], however, a detailed analysis shows that for some interactions individual methods perform better than the consensus network [319].

Co-expression networks and module identification algorithms

Co-expression networks contain genes as nodes and the edges of the network represent significant co-expression levels across the studied data set. An open problem is still how these connexions are to be defined and how an adequate threshold is to be imposed [538]. In differential network analysis two networks obtained using the same algorithm but alternative datasets are compared to identify interactions appearing only under a subset of conditions [196, 236, 539]. For example, such an approach was used to analyse prostate cancer datasets by comparing the networks inferred using C3NET from normal or tumour samples datasets and was able to successfully identify cancer specific interactions [14]. Here, we construct and analyse co-expression networks extracted from the same dataset but using alternative algorithms to identify relevant interactions.

Biclustering or module learning algorithms [313] aim at the identification of functionally related genes showing co-expression patterns. In some cases, such as the cMonkey algorithm [409], additional biological information (databases or sequence analysis) is also considered. The results of the genome-scale reconstruction methods can be displayed and visualized as one network whereas an identified module or bicluster contains a set of functionally related genes that might be differently regulated in different conditions. Therefore, the genes within a module might not form a cluster when considering all the conditions and would remain undetectable for network inference methods. Network inference methods and module prediction algorithms are highly complementary and new information can be obtained by combining the outcome of both approaches.

Analysis of ChIP-seq data

There are experimental methods to directly reconstruct regulatory networks. The knowledge of transcriptional regulatory events and specifically the transcription factors (TF) binding sites can be greatly improved by chromatin immunoprecipitation and sequencing, ChIP-seq. Powerful techniques have been developed to process the sequencing data and to isolate the binding sites from the background noise generated by non-specific sequences, such as statistical tests, use of controls, techniques for signal de-convolution or lag-analysis among others [311]. In the classical model of transcription regulation in prokaryotes a binding site in the promoter region of a gene is linked to a regulatory interaction between the corresponding TF and the gene. However, the collection of ChIP-seq data for *M. tuberculosis* (*Mtb*) hosted in the TB Database (TBDB) [181, 406] shows that not all identified TF binding sites can be linked to regulatory interactions. For example, in some instances the TF binding is much weaker than expected, in other cases this association is complicated by the occurrence of divergons, which are pairs of divergently transcribed operons or genes, or by the presence of binding sites within coding regions. This lack of a one-to-one relationship between TF binding sites and regulatory events can be due to multiple reasons such as non-specific binding, cumulative effects of sites with weaker binding to regulate overall promoter affinity, specific binding generated in non-biological or non natural conditions, false positives as a result of the experimental procedure, or, simply errors in the genome annotation. We propose that it is through the integration with additional data sources, specifically through the integration with expression data, that this challenge can be overcome.

Integration of heterogeneous molecular networks

Each inference algorithm has its weaknesses and strengths and each network holds its own intrinsic value and can be used to gain more specific insights on various aspects of the biological system [156]. In a sense, using multiple algorithms to extract the networks and presenting them to the user is a similar approach to that followed by annotation pipelines, such as Microscope [507], DIYA [475] or BASys [508] among others. These pipelines present the user a list of gene centred information, with different annotation sub-fields, to enable supervised annotation. Integration efforts require a common layout for the data, therefore, we have chosen to represent the available layers of information, such as operon structure, known interactions between genes or proteins, enzymatic activity (metabolic map) or functional similarity, among others through network representations. The networks themselves are represented in a common format, XGMML (eXtensible Graph Markup and Modeling Language), that allows their simultaneous exploration.

In addition to the simultaneous visualization of different networks, additional analysis tools such as motif search and identification, GO-enrichment analysis, and tools to overlay expression data or analyse expression profiles of multiple genes across different conditions can be used. We have developed a pipeline for the generation of co-expression networks that is easily tunable to produce alternative networks (holding their own intrinsic value). The pipeline is based on using the similarity between gene expression profiles to detect associations and contains a higher order extension of the data processing inequality method [48] to reduce the number of redundant or possible spurious links. This pipeline is presented in Figure 2.1.

Through the exploration of the multiple networks, the user gains new knowledge of the biological system, but it can also lead to the discovery of new strategies to integrate the information stored in the networks. These newly gained strategies can then be translated into new pipelines. As result of our exploration, we have developed a pipeline to uncover multiple layers of regulation, presented in Figure 2.2 and another one to analyse ChIP-seq data and assign regulatory interactions to the detected binding sites, presented in Figure 2.3.

Show case: Deployment of the framework to unravel regulatory mechanisms in *M. tuberculosis*

We have analysed regulatory events in *Mtb*. Due to its implications to human health this highly successful pathogen has been extensively studied and there is already a substantial body of information on *Mtb* and its underlying regulatory networks, but still much remains to be learned. We have analysed not only networks extracted from literature [36, 426, 431] but also networks extracted from publicly available databases such as STRING [483], MetaCyc [95], KEGG [259], TBDB [406] and Tuberculist [297]. Expression data from publicly available repositories and corresponding to 287 perturbations, have been used

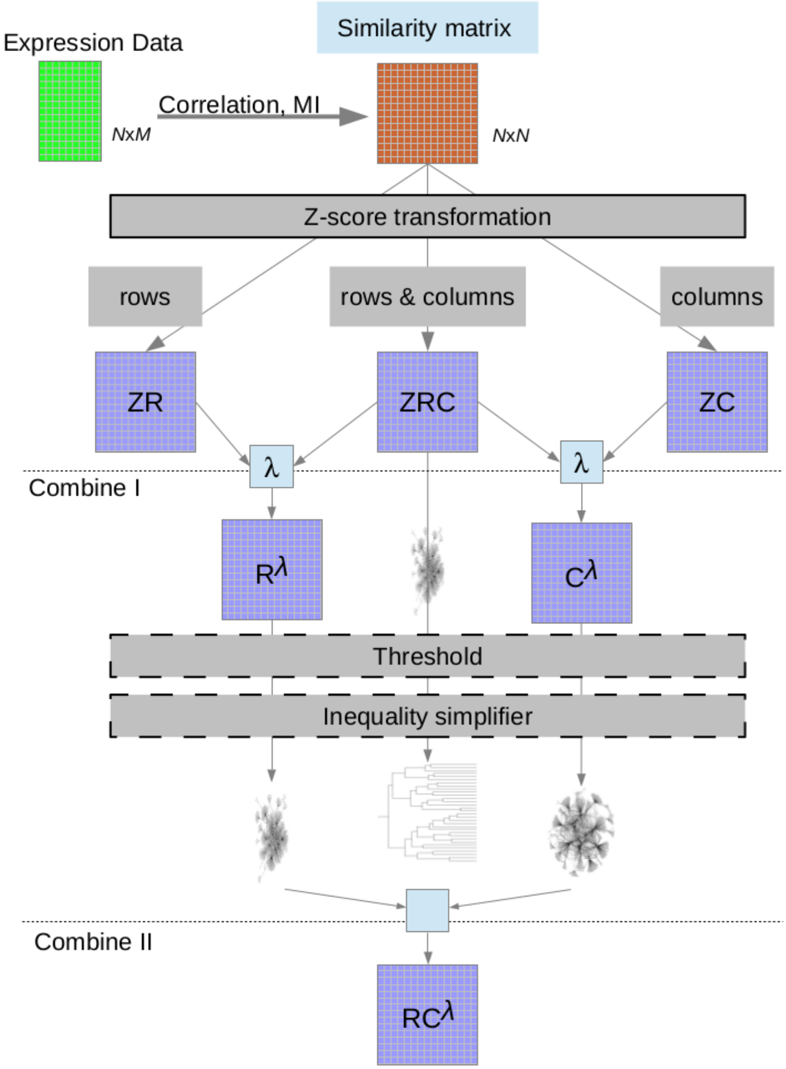


FIGURE 2.1: **Schematic of the pipeline to obtain co-expression networks.** From top to bottom the following steps are applied: (1) calculation of the similarity matrix, (2) z-transformation, (3) Combine I, (4) threshold setting (5) inequality simplifier and (6) combine II. Note that when applying the Inequality simplifier to the ZRC network the result will be a tree.

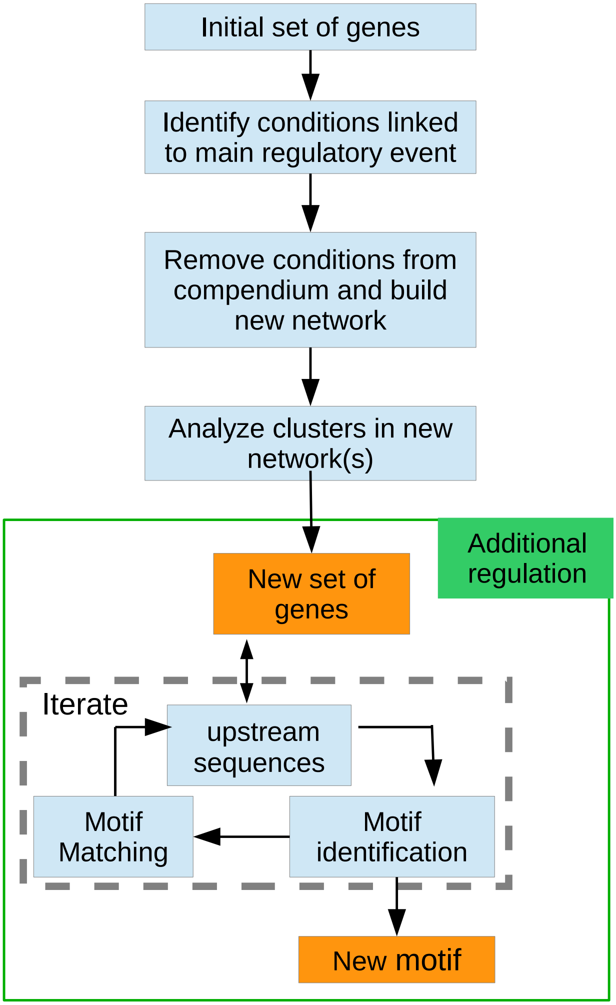

Initial set of genes
Identify conditions linked to main regulatory event
Remove conditions from compendium and build new network
Analyze clusters in new network(s)
Additional regulation
New set of genes
Iterate
upstream sequences
Motif Matching
Motif identification
New motif

FIGURE 2.2: **Pipeline to uncover additional regulatory layers.** Step 1: Identify conditions linked to the main regulatory event for the initial gene set. This can be done using biclustering techniques or by direct comparison with the expression levels of the regulator (if known). Step 2: Build co-expression networks in the remaining conditions. Step 3: Identify the closest neighbours of the selected genes in the new networks. Step 4: iterative round of motif identification/matching to identify the secondary motif and the set of genes with this motif in their upstream regions.

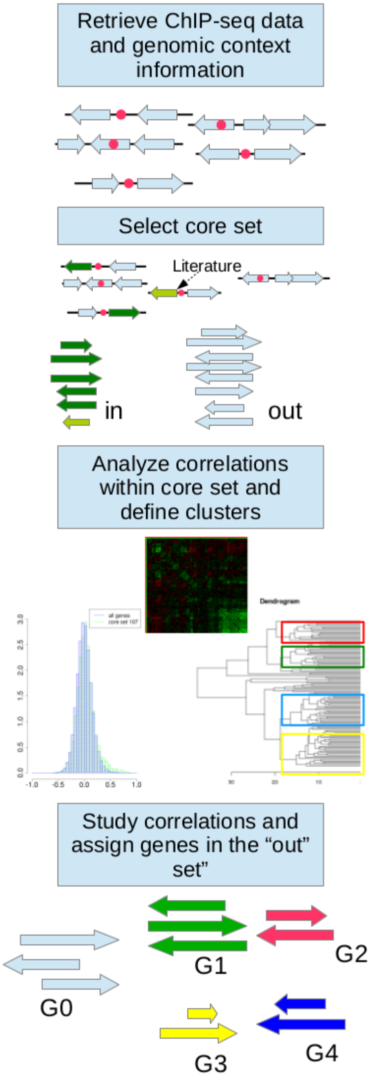

Retrieve ChIP-seq data and genomic context information
Select core set
Literature
in
out
Analyze correlations within core set and define clusters
all genes
core set 107
3.0
2.5
2.0
1.5
1.0
0.5
0.0
-1.0
-0.5
0.0
0.5
1.0
Dendrogram
30
20
10
Study correlations and assign genes in the “out” set”
G0
G1
G2
G3
G4

FIGURE 2.3: **Pipeline to analyze ChIP-seq data.** After the locations of the ChIP-seq binding sites have been retrieved, their genomic context is analysed. A core set is defined by selecting targets with i) literature evidence or ii) a hit in the upstream region of not divergently transcribed genes. The expression levels of these genes are analysed and they are categorized through (bi)clustering. Finally the rest of the putative targets are assigned to these groups (if possible) based on the similarities of their expression patterns.

to generate the co-expression networks and to further analyse ChIP-seq data [181]. We have finally combined the inferred networks with the outcome of biclustering algorithms, to further explain the functionalities of the modules.

We have analysed the regulation of DNA repair systems within *Mtb*, particularly we have focussed on two alternative regulatory mechanisms: the 'RecA-LexA dependent DNA repair system' and 'RecA non dependent regulation'.

We will show that the integration of expression data and its use to guide the exploration of upstream sequences allows to analyse alternative regulatory mechanisms for the same set of genes. In addition, we have explored some of the regulatory mechanisms that allow *Mtb* to survive within the host, such as DevR (DosR) regulon, which is a key element for understanding the dormant state, and the regulation of the response to changes in zinc availability (ZuR regulon).

Methods

Gene expression data

565 two-colour microarrays for *Mtb* (strain H37Rv) were retrieved from the Gene Expression Omnibus Database [45]. 454 of them aimed to capture the effect of 75 drugs targeting metabolic pathways [72, 263] whereas 111 captured stress induced dormancy in the wild-type and in DevR activation genes knockout mutants [131, 228]. We followed Boshoff *et al.* [72] to classify the conditions in the compendium into 14 categories according to the experimental perturbation: (1) Aromatic amides intracellularly hydrolyzed, low pH; (2) mutants of DevR activation pathways; (3) Translation inhibition; (4) Acidified medium; (5) Cell wall synthesis inhibition; (6) Respiration inhibition (except conditions with NO); (7) Nutrient starvation; (8) DNA damage; (9) Transcription inhibition; (10) Iron scavengers; (11) Multiple stress sources applied simultaneously (low oxygen, low pH, glycerol-deprived); (12) Minimal medium (succinate/palmitate as carbon source); (13) not classified; and (14) conditions associated with DevR upregulation. A list of the used datasets is presented in Additional file 1. We applied a common normalization method, loess, to the arrays from each experiment and each of the six independently designed platforms. Linear models were constructed to consider biological and/or technical replicates (when available); within-array replicate probes were averaged; and an additional between-array quantile-quantile normalization was performed to ensure the comparability between experiments. These manipulations were performed using the R limma package [464]. A common locus tag format was introduced and missing values were filled up using knn-imputation from the impute R package [218] with $k = 3$ and eliminating genes and experiments with more than 30% and 50% missing values respectively. The resulting compendium contained information on 4223 open reading frames across 287 different conditions (175 steady state situations and 100 conditions within 30 time series).

Biclustering

We adapted the original cMonkey R code to *Mtb* data [409] and we increased the number of biological information sources that the algorithm can consider. For the biclustering process, we selected a subset of the interactions present at the STRING database [483]. We selected specifically interactions obtained based on co-occurrence of linked proteins across different species, on curated databases, and on their association in the abstracts of scientific literature. From ProLinks [73] we collected interactions obtained by the phylogenetic profile method. In addition, we included the network based on the similarity among the annotated GO terms (for the ontologies 'biological process' and 'cellular component'), which was constructed using the Sleipnir library [233]. Upstream sequences for *Mtb* genes were retrieved using RSAT [497] and used for the motif detection and motif matching steps. For the automatic biclustering algorithm, we used the 1000 bp region upstream of the translation start site, avoiding overlapping with upstream neighbour genes when present. In the initial rounds we used a reduced matrix where only the leaders of the operons, as defined by Roback and co-workers [414], were kept. Multiple runs of the algorithm were performed, which used the default parameters except for: initial size of seed clusters (10); number of iterations (1000); maximal number of clusters (300). We obtained a set of 1527 biclusters. The Jaccard similarity coefficient between each pair of biclusters was computed (number of shared genes between both biclusters over the number of different genes in both biclusters). To reduce the redundancy in the set of biclusters, we merged the pairs that showed a Jaccard similarity higher that 0.7. The biclusters in the merged set were enlarged by adding genes based on the predicted operon structure combined with the expression level measurements. This new set was used to seed the biclustering algorithm in two subsequent optimization rounds one biased towards the detection of concurrent motifs in the upstream regions and a second one biassed towards the identification of sets linked to highly related biological processes (GO terms). We performed an additional manual merging step, that also considered the similarity between the detected motifs using the matrix comparison tools from RSAT [497] and we obtained a final set of 76 biclusters. Expression plots for these biclusters are in Additional file 2.

Pipeline for the generation of co-expression networks

The pipeline to generate co-expression networks is presented in Figure 2.1. Departing from an $N \times M$ matrix containing the expression profile of N genes in M conditions, we construct a symmetric $N \times N$ similarity matrix S. The pipeline allows to choose between correlation (Pearson, Kendall and Spearman) and MI as similarity measurements. MI is computed using an estimator based on the entropy of the empirical probability distribution with initial data discretization into n (default = 10) equal sized bins from the Bioconductor package minet [337]. In the second step, a z-score transformation is performed on the distribution of the scores from the similarity matrix. The transformation is done by rows, to obtain the ZR matrix; by columns, to obtain the ZC matrix; or by both rows and columns simultaneously, to obtain the ZRC matrix. The z-score transformation allows for each possible interaction to be weighted regarding the background of interactions in which each member of the interacting pair is involved [161]. Therefore, in the ZC case each element S_{ij} becomes ZC_{ij}, which is the z-score of S_{ij} regarding the distribution S_{i1}, S_{i2}, …, S_{in}. In the ZR case each element S_{ij} becomes ZR_{ij}, which is the z-score of S_{ij} regarding the distribution S_{1j}, S_{2j}, …, S_{nj}. In the ZRC case each element S_{ij} becomes

$$\sqrt{(ZC_{ij}^2 + ZR_{ij}^2)/2}. \tag{2.1}$$

In the following step the matrices are combined into two new matrices: C^λ and R^λ:

$$C_{ij}^\lambda = max(ZC_{ij}, \lambda \cdot ZRC_{ij}) \tag{2.2}$$

$$R_{ij}^\lambda = max(ZR_{ij}, \lambda \cdot ZRC_{ij}) \tag{2.3}$$

with λ (default = 1) a positive real number and i and j denoting the rows and columns of each matrix. λ allows to fine-tune the results, since higher values of λ will lead to C^λ and R^λ matrices that are similar to each other and to ZRC. Considering $\lambda = \sqrt{2}$ means that, for those cases where one of the elements in (2.1) is zero, both λ ZRC_{ij} and ZC_{ij} will be identical. However, using the default $\lambda = 1$ will ensure that when the values of ZC_{ij} and ZR_{ij} differ then the highest one will be selected either through ZC_{ij} (or ZR_{ij}) or ZRC_{ij}.

In the following step a threshold can be applied to remove interactions with low weight (and therefore low likelihood). Any value below the threshold is set to 0. This step produces a more sparse network, which might be needed to obtain a neat visualization. The default threshold value is chosen so that the number of non zero edges is equal to a predefined value (default = 10000).

Afterwards the inequality simplifier can be used to remove, possibly spurious, links from the network. The inequality simplifier is an extended version of the data processing inequality (DPI). In simple terms, the DPI states that given two interdependent random variables A_1 and A_2 and a third one A_3 that only depends on one of them, for example A_2, then A_1 cannot contain more information about A_3 than A_2 does. This statement is mathematically represented by the following inequality among MI values

$$MI_{A_1,A_3} \leq \min(MI_{A_1,A_2}, MI_{A_2,A_3}). \tag{2.4}$$

The DPI must hold whenever A_3 does not depend on A_1, so it can be used to remove spurious interactions from the network [48], possibly caused by feed forward loops or mutually dependent regulators. The spurious links are removed according to:

$$\textit{if}\, MI_{A_1,A_3} \leq MI_{A_1,A_2} \;\;\&\;\; MI_{A_1,A_3} \leq MI_{A_2,A_3}, \text{ then, set } MI_{A_1,A_3} = 0. \tag{2.5}$$

The DPI can be extended to higher-order interactions, therefore allowing a recursive implementation [245]. The DPI is derived from the triangle inequality satisfied by any metric or distance, such as the MI or Kendall rank distance. Our extension to remove spurious edges by considering higher order interactions contains two parts: the first part is the identification of alternative pathways connecting the same nodes and the second part is to apply the inequality simplifier to decide whether a link is spurious and should be removed. The first part is based on Dijkstra's shortest route algorithm [143]. Here, instead of searching for the shortest route between two nodes (e.g. genes), we use it to find the alternative path between those two nodes with the best throughput. The throughput of a path is defined as the value of the edge with the lowest throughput (similarity) value. Given two connected nodes, the link between them is considered spurious and removed if an alternative path is found with a higher throughput value. For a given pair of nodes, A_i and A_j, $(i < j)$, so that the best alternative path of length $(n+1)$ passes through A_1, $A_2, \ldots,$ A_{n-1} and A_n, the following rule (inequality simplifier) is applied:

$$\begin{aligned} \text{if} MI_{A_i,A_j} \leq \min(MI_{A_i,A_1}, MI_{A_1,A_2}, \ldots, MI_{A_{n-1},A_n}, MI_{A_n,A_j}) \\ \text{then, set } MI_{A_i,A_j} = 0, \text{ where } n \in \{1, \ldots, N-2\}. \end{aligned} \tag{2.6}$$

This rule is applied for each possible pair of nodes and for each possible value of n. The alternative pathway joins A_i and A_j and does not contain any cross-linking, so that therms such as A_2A_4 or A_1A_3, do not need to be considered. This is a consequence of using the path with the best throughput so that no triangle inequality of the metric is required to derive this property.

Figure 2.4 shows how the inequality simplifier acts to remove, possibly spurious, links from the network when applied to higher order terms. While applying this rule, an additional $N \times N$ matrix, T, is built to keep track of how many links a particular edge has caused to be removed. In this case, the links A_i, A_1, A_1A_2, $A_2A_3, \ldots,$ $A_{n-1}A_n$ and A_nA_k have caused the removal of link A_iA_j, therefore, in the T matrix, the elements T_{A_i,A_1}, T_{A_1,A_2}, T_{A_2,A_3},$\ldots$, T_{A_{n-1},A_n} and T_{A_n,A_j} are increased by $1/(n+1)$.

Finally the two networks C_{ij} and R_{ij} are combined into one final network RC_{ij}:

$$RC_{ij}^{\lambda} = \max(ZC_{ij}, ZR). \tag{2.7}$$

The presented pipeline can be used to obtain different outputs, such as the RC or the ZRC networks and their variants through the application of thresholds or the inequality simplifier. The final output is a square matrix that represents a network through a weighted adjacency matrix. This adjacency matrix

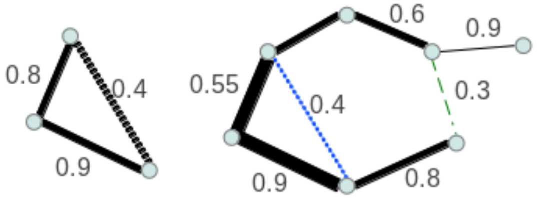


FIGURE 2.4: **The inequality simplifier.** The similarity values among the nodes (genes) connected by the different edges have been indicated. Dotted lines represent the spurious links removed by the inequality simplifier. *Left:* application of order two, which is equivalent to a direct application of the DPI. *Right:* higher order application, both dotted lines are removed. The DPI would only remove the blue dotted line.

is finally exported into a tabular format that can be imported into Cytoscape [463] to obtain a graphical representation. The weights in the adjacency matrix or the T matrix produced by the inequality simplifier can be imported as edge attributes and used to set the thickness of the edges. Finally, a force directed algorithm can be used to generate the appropriate layout and the graphical representation of the network, which then can be exported to an XGMML file.

Applying the inequality simplifier to a symmetric matrix, such as ZRC, removes any possible loop in the network, therefore it results in a forest graph as depicted in Figure 2.1. Applying the inequality simplifier to an initial matrix containing around 3000 nodes takes at most five minutes in a standard desktop computer (2.80GHz Intel machine).

The networks used for our exploration of gene regulation networks in *Mtb* were the result of applying the inequality simplifier to ZR, ZC and ZRC (obtained with $\lambda = 1$). Belcastro *et al.* [53] have shown that from the total number of possible interactions a selection of 5% (4*105 for a network with 4000 genes) would be a sensible choice according to the assumption that biological network are sparse. Here, one of our goals is to obtain a clear visualization of the network, therefore, we have chosen an even lower number (10000) so that the number of nodes included in the network is as high as possible without having a too crowded visual representation of the network. An additional threshold was imposed so the number on non zero edges of ZR, ZC and ZRC networks is equal to 10000. The ZRC derived network contained 2293 nodes and 10000 edges whereas the mixed network contained 2693 nodes and 5898 edges.

These manipulations were implemented in R and Java, the corresponding scripts can be found, together with an example data set in Additional file 3.

Additional networks and analysis tools

Metabolic networks. The nodes in the metabolic networks are of two different types: reactions and metabolites, and additional information, such as the stoichiometry, directionality and the catalysing proteins of the reactions, also has to be stored. We kept this information by using a combination of two file types: a SVG (Scalable Vector Graphics) file that contains the graphical representation of the networks and an RDF (Resource Description Framework) file containing the additional information (see Additional file 4 for technical details).

Other networks. A network formed by multiple disjoint interlinked clusters was generated using the predicted *Mtb* operon structure [414]. In addition, the information stored in the STRING database [483] was used to generate networks based on: known protein-protein interactions; distance of the genes in the genome and that of orthologous genes in other species; co-occurrence in other species; orthologous genes being fused together in other species; association in the abstracts of scientific literature; and other databases, such as MetaCyc [95] and KEGG [259]. To identify regions duplicated in the genome or homologous genes, a network was built by considering the sequence similarity between each pair of genes. The similarity was computed using Megablast [344] and an edge was created for matches with E-values lower than of 10^{-10}.

Sequence analysis tools. We have used MEME [34] for motif elicitation within a set of genes and FIMO [205] to find occurrences of the chosen motif. A post processing step ranks, using the FIMO q-value, the genes (matching upstream region) for each identified motif (MEME). An iterative cycle of motif identification and motif matching can be established to further refine each motif. The computational time required by each iteration greatly depends on the length of the upstream sequences, which has to be selected for each particular analysis. For the selected show cases the motifs accumulated in the 200 bp region upstream of the translation start site.

Gene ontology (GO) enrichment analysis on selected set of genes was performed using a hypergeometric function to model the probability density, as implemented in the GOHyperGAll R function [230]. The GO annotation for *Mtb* was obtained from the UniProtGOA database [144], the Mtb-GOA database (`http://www.ark.in-berlin.de/Site/Mtb-GOA.html`), TBDB [406] and the more recent re-annotation of the *Mtb* genome [146].

Operons. We developed a putative operon extension tool which is only based on the gene orientation, so that genes that are downstream of the selected gene (or genes) and with the same orientation are added to the putative operon until a gene with the reverse orientation is found. The user is then free to decided whether the expression data sustains the extension of the operon. This approach allows to easily combine expression information with genomic context information.

Venn diagrams have been created using the utilities from the VennDiagram R package [103].

Expression plots. Generated with a solid line representing the mean expression of the selected genes in the conditions included in the compendium and dots marking the conditions in the bicluster where the genes show high correlation (bicluster). The classification of the experimental conditions, previously mentioned, was included in the expression plots through an horizontal colour line that allows to quickly associate the behaviour of the genes with the type of perturbation. To select the conditions (if any) in which the selected genes show co-expression there is an initial step where all conditions are included in the bicluster, then an iterative loop starts were each condition is removed at a time and the new correlation values are computed, finally the condition leading upon removal to the highest correlation values in the remaining set, is removed from the set. This process is iterated until the correlation in the remaining set of conditions is higher or equal to 0.8.

Discovery of additional regulatory layers

Whenever two alternative regulators regulate the expression of genes in a given set, it might be that one overshadows the detection of the other. We have developed a work flow to analyse these events (shown in Figure 2.2). Given a set of genes under control of two regulatory events, the first step is to identify the conditions were the first event is active or taking place. How these conditions are identified depends on the particular example. One way would be to identify conditions with up/down regulation of the corresponding transcription factor. To identify conditions were *recA* expression is not regulated by the RecA/LexA mediated mechanism we built a model linking the time dynamics of *recA* to *lexA* expression levels using the Inferrelator algorithm [70]. The output of the algorithm is a prediction or fit of *recA* levels based on *lexA* levels (see Additional file 5). Those conditions with a poor agreement between the measured and fitted levels are the conditions (most likely) without LexA mediated induction of *recA*.

Once the new set of conditions has been selected, the following step is to reconstruct the co-expression network(s), using only the expression data corresponding to this subset of conditions and analyse the location of the original set of genes in the newly built network(s). Furthermore an iterative round of motif identification and motif matching can be run to identify a possible motif in the upstream region of the cluster. This iterative round can be performed either using cMonkey [409] (restricting the expression data to the selected conditions) or manually using alternatively MEME [34] and FIMO [205].

Analysis of ChIP-seq data

Publicly available ChIP-seq data were obtained from TBDB [181]. We have developed the workflow, presented in Figure 2.3, to analyse these data. The ChIP-seq data had already been analysed with a peak calling algorithm. For each of the considered TFs a list of predicted targets was obtained from TBDB, together with information about the peak location. A reduced list, or core set, was obtained by selecting those targets that show a hit in their adjacent upstream intergenic regions. An additional filtering was done to select only those hits where the peak is not flanked by divergently transcribed genes/operons. Additionally, the genes for which literature evidence supported the regulatory interaction with the TF were included in the core set. In the following step, the expression levels of the genes in the core set were analysed and the matrix of correlations across the conditions in the expression compendium was computed. The appearance of negatively correlated genes in this set is a signal of a dual repressing/activation function of the regulator and therefore two (or more) subsets can be defined by hierarchical clustering of the set. This process can be further enlarged to encompass additional groups that would be linked to alternative regulatory mechanisms or to the effect of additional TFs. Once this/these group(s) were defined, the rest of the putative targets predicted by ChIP-seq were assigned to either one of these groups based on their average correlation with the members of the group (0.7 threshold).

Topological overlap

The topological overlap between two genes i, j in an unweighted network is defined as [404, 535]:

$$t_{ij} = \begin{cases} \frac{|N(i) \cap N(j)|}{\min\{|N(i)|,|N(j)|\}+1-a_{ij}} & i \neq j \\ 1 & i = j \end{cases}, \tag{2.8}$$

where $N(x)$ is the set of direct neighbours of gene x (excluding itself); $|x|$ represents the number of elements in set x and $a_{i,j}$ is the adjacency matrix of the network (1 if there exists a link between genes i and j and 0 otherwise). The topological overlap is bounded between 0 and 1. Two genes will have high topological overlap if there exists a connection between them and if they are connected to the same group of genes. For weighted networks, $a_{i,j}$ represents the weight of the interaction between genes i and j and takes continuous values between 0 and 1. In these cases, the previous formula can be generalized to [538]:

$$t_{ij} = \frac{\sum_k a_{i,k} a_{k,j} + a_{i,j}}{min\{\sum_k a_{i,k}, \sum_k a_{j,k}\} + 1 - a_{i,j}}. \tag{2.9}$$

The topological overlap of a group of genes was defined as the average of their mutual topological overlap, which was computed using the R WGCNA package [286]. Only groups with more than 5 genes were considered. An empirical p-value for these scores was calculated by randomly sampling (10000 times) gene groups of the same size in the respective network.

Network visualization

We have developed a visualization tool that allows the simultaneous visualization of networks in XGMML format and the sharing of identifiers between them. Technical characteristics of this tool are available in Additional file 4.

Results

Co-expression networks

We have developed a pipeline to generate co-expression networks that allows for a myriad of possible networks. Choosing a subset of them highly depends on the available data and the process the user wants to explore, since different clusters appear in each of them. This reflects the inherent modularity of biological networks and the dependency of regulation on the chosen conditions.

The pipeline presented in Figure 2.1 includes some already well established and tested methods for genome scale network inference. For example, the similarity matrix computed using either correlation or MI is commonly used to analyse gene expression data [128], but also other methods are contained in the pipeline. When working with mutual inference as a similarity measure the matrix denoted ZRC corresponds to the output of the CLR algorithm [161]. Additionally, applying the inequality simplifier only up to order two on a MI matrix amounts to using the DPI to prune spurious interactions from the networks, which is a key element of the ARACNE algorithm [48].

A comparative analysis of methods to reverse-engineer transcriptional regulatory networks was done using the results of the DREAM5 challenge. In this challenge the teams had to reconstruct genome-scale transcriptional regulatory networks from expression data. The networks proposed by the different teams, were evaluated through their comparison with a gold standard, a set of experimentally verified regulatory interactions in the target organisms. In addition, Marbach *et al.* [319] constructed a consensus network integrating the predictions from the multiple methods. Whole network performance estimators such as Area Under Receiver Operating Characteristic Curve (AUROC) or Area Under Precision Recall Curve (AUPR) show that over the entire network the consensus network outperforms individual methods due to their inherent complementarity. However, for some interactions individual methods perform better than the consensus network (see Additional file 6A). The loss of information when building the consensus network does not affect equally the different transcription factors (see Additional file 6B). For example, for PurR or LexA, a significant fraction of the total number of known interaction is better recovered by the individual methods than by the consensus network.

To evaluate the performance of our algorithm we have generated co-expression networks corresponding to the synthetic, *Escherichia coli* and *Saccharomyces cerevisiae* datasets used in the DREAM5 challeng. For each dataset, two networks (ZRC and mixed) were built (see Methods section). In the challenge the goal was to identify links between regulators and target genes. In coexpression networks no special emphasis is done on the regulators. To evaluate the networks we have calculated the topological overlap [404, 535] of the known targets of the transcriptions factors (also provided in the challenge). The topological overlap measures how similar the neighbourhoods of two genes are [538]. For 95% (64 out of 67) of *E.coli* TFs considered (those with more than 5 experimentally verified targets) the topological overlap of the target genes in the ZRC network is significantly higher than for the overall network (see Figure 2.5), which means that they form cohesive modules in the network. The relative number of cohesive clusters identified for the other datasets (yeast and synthetic set) are lower (57% and 40% respectively, see Additional file 7) Overall the ZRC networks maintain a higher degree of cohesiveness than the mixed networks (89%, 31% and 32% for the E.coli, yeast and synthetic datasets).

Systematic analysis of regulation in *Mtb*. Integration and simultaneous exploration of heterogeneous networks

We considered co-expression networks obtained with different approaches. Additional information is represented through networks, such as information on operons, GO annotation similarity, data base stored knowledge and sequence homology among others.

This integrative approach allows to systematically explore functional modules in the network and it is highly complementary to existing bi-clustering and other module identification methods. Even when additional biological information is included, biclustering methods have to face the challenge of interpreting the function of each groups of genes. Therefore, a general approach to detect and understand functional modules in a given organism is to simultaneously explore the location of the genes in a given bicluster in the co-expression networks. We have done so to analyze the output of the cMonkey algorithm [409] on *Mtb* data. We have run multiple instances of the algorithm to bias the search towards the detection of both putative co-regulated and functionally related sets of genes. The obtained bi-clusters are presented in Additional file 2. To further investigate these biclusters, we have projected them in the multiple networks and genes proceed to their in-depth analysis. An example of how this is done is shown in the following section when we analyse the regulation of DNA damage repair systems in *Mtb* by LexA.

In a given co-expression networks it might very well happen that the effect of one regulator on a given set of genes overshadows the identification of other regulators. However, these mechanisms can be uncovered when comparing different networks generated under different experimental conditions. We have developed the pipeline presented in Figure 2.2 to uncover additional

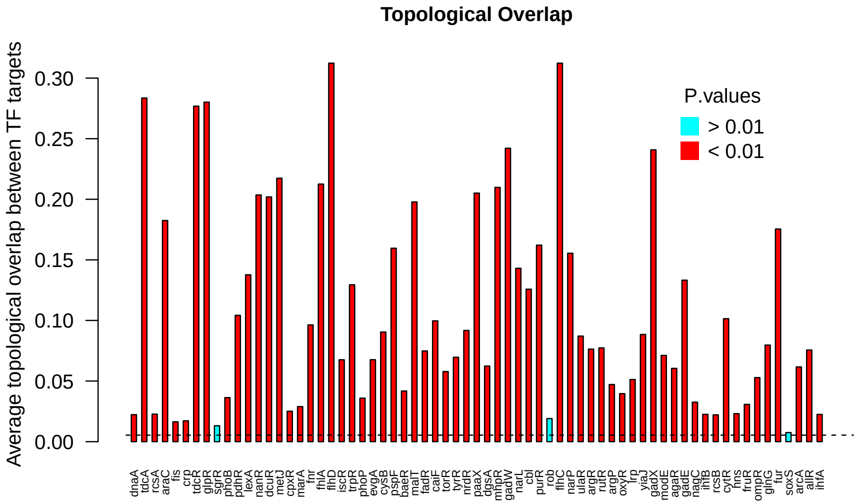


FIGURE 2.5: **Topological overlap of TF targets in the *E. coli* ZRC co-expression network.** The ZRC co-expression network was reconstructed using our pipeline using *E. coli* expression data from the DREAM5 challenge [319]. Only the 67 TF with more than 5 experimentally verified targets (in the gold standard) were considered. Dashed line represents the average topological overlap in this network (0.0053).

layers of regulation based on the assumption that regulation proceeds differently upon different perturbations, therefore some regulatory events will only be detectable in subset of conditions. This pipeline is only useful when the user suspects or has prior knowledge of an additional regulatory interaction affecting the same set of target genes. Therefore, we have used it to further explore the regulation of DNA repair systems in *Mtb*, where additional regulators are known to exist [183].

In addition, our approach allows to systematically explore and interpret additional data such as ChIP-seq data, through the pipeline presented in Figure 2.3. Among the available data we have chosen to focus on one of the key subsystems key for *Mtb* survival in the host, the regulation of the response to hypoxia and the induction of dormancy program via DevR.

Furthermore, the simultaneous network visualization, allows the exploration of common trends in the networks. Instead of focussing on clusters of genes appearing in the co-expression networks and how these clusters are reflected in, for example, the network liked to the GO biological process annotation. The alternative approach can be taken and for each set of genes linked to a particular GO or COG category we can trace back their location in the network. This analysis can point to interesting effects. For example, we explored the genes coding for ribosomal proteins and in the different co-expression networks these genes appear to form a highly interlinked cluster. However, in all versions of the networks, the *rpmB2-rpmG1-rpsN2-rpsR2* operon (*Rv2058c-Rv2055c*) formed by genes coding for ribosomal proteins (S18-S14-L33-L28) appear as a separated set (see Additional file 8). The combination of this information with the network of interactions extracted from literature [431] led us to the analysis of the zinc uptake regulator, ZUR, and its targets.

DNA repair systems in *Mtb*

RecA-LexA dependent DNA repair system: the SOS box

LexA is a repressor known to be involved in the control of mechanisms for DNA repair. Under neutral conditions LexA, dimerizes and binds to the SOS box, repressing its own expression and of other genes related to DNA repair. The consensus sequence of the SOS box for *Mtb* had been identified as the palindromic motif TCGAAC(N)^{4}GTTCGA [129]. Upon DNA damage, RecA binds single stranded DNA (ssDNA). The complex RecA-ssDNA, stimulates autocatalytic cleavage of LexA, so genes repressed by LexA are induced. Acidic conditions trigger a similar response, since LexA can no longer dimerize, effectively preventing it from binding its target DNA sequence [88].

The iteratively mapping of the genes in each of the biclusters presented in Additional file 2 to the multiple co-expression networks, showed that bicluster 36 is a 13 gene module that shares many of the characteristics of the LexA regulon. To further investigate this bicluster, we proceed to analyse the location of these genes in the multiple networks and to identify genes that might be linked to them through the systematic analysis of their expression patterns (see Figure 2.6 A and B). For each of the candidates to be included in the regulon we combined information from the different information layers: literature, databases, functional annotation, and upstream sequence among others. We were able to identify a total of 28 genes putatively in the regulon, listed in Additional file 9: Table S1. This includes the 16 genes reported by Davis *et al.* [129] plus 12 additional ones, resulting in an effective enlargement of the regulon size by 75%, (see Figure 2.6D). Based on this extended regulon, we identified for the SOS box in *Mtb* a more specific motif: MKWMTCGAAMRYWTGTTCGA (depicted in Figure 2.6C).

To verify our predictions about the 12 genes not previously assigned to this regulon, we compared our predictions with the LexA binding regions identified by Smollett *et al.* [462]. In that work ChIP-seq analyses of LexA binding sites was complemented with experimental measurements of gene expression upon DNA damage induced by mytomycin C. The 12 genes that we assign to

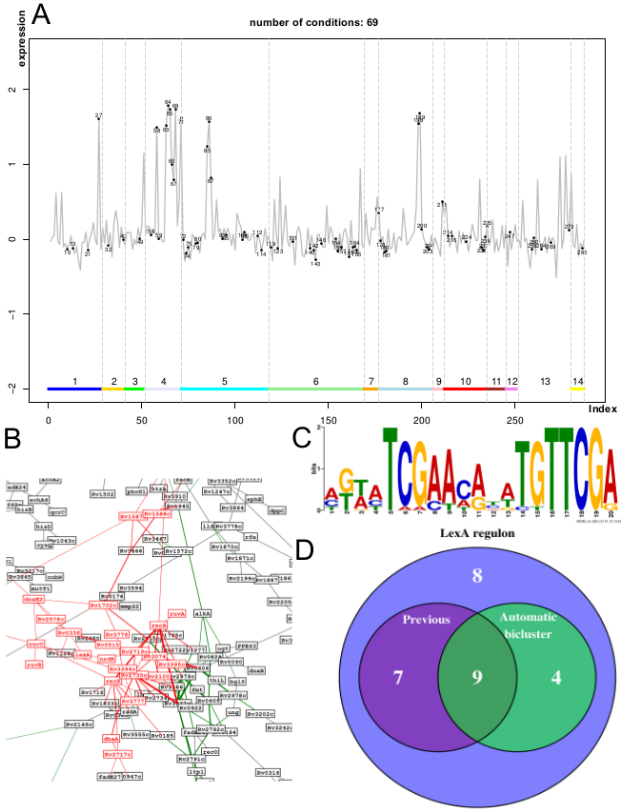


FIGURE 2.6: **LexA regulon. A)** Plot of the average expression level of the members of the LexA regulon across the different conditions. Red dots mark conditions with high (>0.8) correlation between the genes in LexA regulon. The horizontal bar and its different regions indicated by numbers refer to the classification of the conditions as described in Materials and Methods. High expression levels are observed in conditions corresponding to low pH or UV light. **B)** Clusters of genes involved in DNA repair mechanisms in the co-expression network (obtained from the combination of R^{λ} and C^{λ} with $\lambda = \sqrt{2}$). Genes regulated by LexA are marked red. **C)** Refined LexA identified binding motif, positions 14 and 15 were previously non specific. **D)** Number of genes identified to be regulated by LexA. *Previous* indicates genes previously reported in the literature as LexA regulated [129], whereas *Automatic* refers to the genes initially identified by the automatic biclustering algorithm.

the LexA regulon contain a site in their upstream regions where LexA binding was detected. In addition, these genes also show dis-regulation in DNA damaging conditions. It is important to stress that the results from Smollett *et al.* were used in our analysis only to verify our predictions. Therefore, we can conclude that we have successfully reconstructed the regulatory interactions of LexA.

Davis *et al.* identified a single SOS box in the upstream region of the divergently transcribed genes *whiB2* and *fbiA*, however none of them were listed as likely to be regulated by LexA. The analysis of the expression profiles of these two genes, *fbia* and *whib2* led us to conclude that *fbiA*, a probable 2-phospho-L-lactate transferase involved in coenzyme F420 biosynthesis, does not belong to the regulon, since no significant expression changes are observed upon conditions related with induction of the LexA regulon (DNA damage or acidic pH). The expression pattern of *whiB2* gene shows anticorrelation with the rest of the genes in the LexA regulon in all the conditions in our compendium that show upregulation of LexA. Therefore, we concluded that LexA regulation of *whiB2* expression proceeds through a different mechanism than the previously described so that the dimerized form of LexA acts as an inductor of *whiB2* expression. This has also been independently confirmed in experimental conditions were mytomycin C was added to the medium [462].

WhiB2 has been hypothesised to be involved in the regulation of cell division [401]. Functional analysis of the genes co-expressed with *whiB2* further supports the idea that WhiB2 is involved in the regulation of cell division. We also found that not only *Rv2719c*, as reported by Davis and coworkers, but the complete *Rv2719c-nrdR-Rv2717c* operon is under the control of LexA. Information from the STRING database [483] (gene neighbourhood and co-occurrence) allows us to functionally link *Rv2717c* with DNA repair and cell wall synthesis; NrdR is involved in the control of the synthesis of dNTP needed for DNA replication and/or DNA repair [346]; and the cell wall hydrolase *Rv2719c* is involved in suppressing the cell cycle by altering localization of FtsZ rings [99]. Therefore, our analysis has unveiled that repression of *whiB2* and induction of the *Rv2719c-nrdR-Rv2717c* operon are the LexA regulated mechanisms to temporary arrest cell division upon DNA damage.

Furthermore, through the re-annotation of the newly discovered LexA targets, we can identify a putative functional mismatch repair (MR) system. No MR system has been previously identified in *Mtb*. A typical bacterial MR system contains: the mismatch-recognition protein MutS that contains an HNH endonuclease domain; MutH, a nicking endonuclease; and MutL, which acts as a scaffold between these. The MR system additionally contains a DNA helicase; a DNA exonuclease to remove the mismatching nucleotides; and a DNA polymerase together with a DNA ligase to repair and ligate the created gap [324]. The hypothetical MR system that we have identified is formed by: i) some of the 10 HNH endonuclease domain containing genes that belong to the 13E12 family, ii) the Holiday junction DNA helicases RuvA and RuvB together with RuvC, a crossover junction endo deoxyribonuclease, and iii) the DNA polymerase DnaE2, together with ImuA/B that are essential for its function

[520]. The existence of this repair system, under the control of LexA, would solve the apparent inconsistency between the low mutation rates in *Mtb* and the absence of an MR system [148].

Additional regulation: RecA independent DNA repair system

RecA is key to the correct regulation of the LexA regulon and it is also LexA regulated. However, additional DNA repair mechanims have been described in *Mtb*, particularly the RecA non-dependent (RecA_ND) DNA repair system [183], that also regulates recA expression in a LexA independent manner. To analyse this additional regulatory layer, we have used the pipeline in Figure 2.2.

The conditions not linked to the main regulatory event are those 93 where no relationship was found between *recA* and *lexA* levels (see Additional file 5). Using these conditions we built a new co-expression network and afterwards we compared the original and the newly built networks. The members of the LexA regulon appear in the original network as a tight cluster, however, in the second network only a subset of them appear clustered. We selected the genes in this small cluster and proceed to an iterative round of motif identification and motif matching, to finally identify the genes regulated by the RecA_ND mechanism. Our approach does not allow us to identify the regulator of the set, but previous studies point to ClpR (Rv2745c) [519].

We have verified our predictions for the identified motif and the list of targets genes, (see Additional files 10 and 11) through comparison with literature data, since they match those previously described by Gamulin et al. [183] for the RecA_ND DNA repair system. However, there are two striking differences between our results and those previously reported by Gamulin *et al.* We don't find that *sigG* responds to DNA repair, which matches the result reported by Smollett *et al.* [461], on the other hand, we do find another sigma factor, *sigC*, that seems to be regulated in response to DNA damage. The list of genes in the regulon, show that in this case, regulation of cell cycle arrest upon DNA damage is not linked to *whiB2*, but only to the *Rv2719c-nrdR-Rv2717c* operon.

Hypoxia and induction of dormancy program: DevR regulon

One of the main characteristics of *Mtb* is its ability to switch to a non replicating or 'dormant' state that allows it to survive for a long time within the host and renders it less susceptible to antibiotics. The environment inside the host is hypoxic and it might have high concentration of CO or NO released by the host macrophages. Under either low oxygen concentrations or high concentrations of NO or CO, the heme iron from the kinases DevS (DosS) and DosT becomes ferrous, the kinases become CO or NO bound and they get activated. In the active form, DevS and DosT autophosphorylate and induce phosphorylation of DevR, which in turn, can bind its DNA recognition sequence and induce expression of the DevR regulon resulting in the activation of the dormancy program [381].

All the options to build the co-expression networks that we have explored, result in a tight cluster for the known members of the DevR regulon. This is to be expected since dormancy and DevR regulon induction have been an active research topic in the *Mtb* field. From the 287 distinct conditions present in the expression compendium, almost 10% of them (23) are conditions associated with expression of DevR regulon. However, the fact that these genes always appear forming a tight cluster, points to the absence of additional regulatory elements that might cause a differential expression of some members of this regulon.

We have selected this regulon to validate our methodology for ChIP-seq data analysis. ChIP-seq data corresponding to over-expression of DevR were obtained from TBDB [181]. The processed results for DevR contain 475 detected peaks, that correspond to 622 genes that could be possibly regulated by DevR, although in our compendium expression data for only 605 of them were available. We have analysed this dataset following the methodology shown in Figure 2.3, and defined a core-set containing 107 genes. The analysis of the correlations in the expression profile among this set as compared to the overall distribution indicates a common regulatory influence over the selected genes, as shown in Figure 2.7A. Further analysis of the genes in this core set and their behaviour across the conditions in the compendium lead us to identify five distinct groups of genes within the identified targets (see Figure 2.7C and Additional file 12). In four of these groups there is a high correlation among the genes whereas no clear pattern can be identified in the expression of the genes in the fifth group. One of these groups, that from now on we will refer to as the DevR regulon, contains 64 genes that were identified by ChIP-seq, has an expression pattern consistent with the previously described DevR regulon, and 37 of these genes have been previously identified as DevR regulated genes (Figure 2.7B). Additionally, we have found among the list of targets from TBDB, 7 genes that have been previously reported as DevR regulated [36, 46, 98, 381] however their expression patterns suggest that either they are not regulated by DevR or there is a secondary regulatory event altering their expression levels, therefore it is arguable whether they can be included in the regulon. A detailed list of members of this regulon is provided in Additional file 13: Table S2. The genes in this table have been assigned to functional categories: cell wall, transport elements, anaerobic respiration, translational machinery, regulatory elements and elements related to stress response. These six elements are linked to some of the main changes observed during growth arrest and dormancy, such as the changes in the cell wall, the arrest in protein synthesis and the adaptation to a hypoxic environment with reactive nitrogen species [193, 486].

Interestingly, we found a faint link between the type VII secretion system (Esx-3) genes *Rv0282-Rv0290* and DevR. Although the correlation analysis shows that they are not members of DevR regulon, we indeed find that, in a reduced set of 27 conditions they show a high (0.7) correlation with DevR regulon (see Additional file 14). These genes are also known to be regulated by Zur (Zinc uptake regulator) [312] and IdeR (Iron dependent regulator) [417]

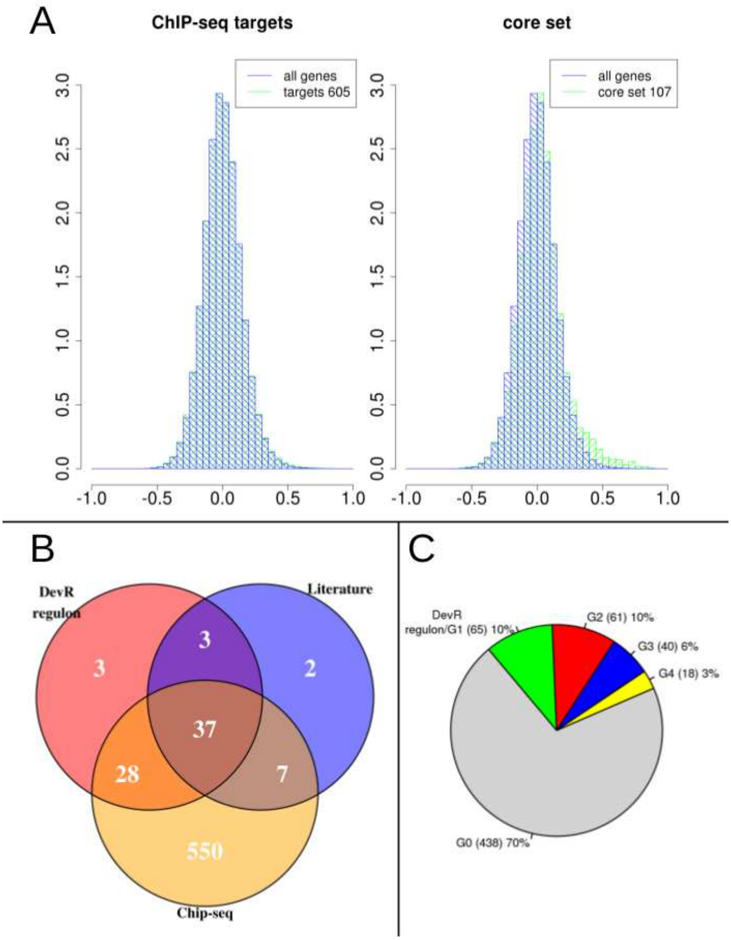


FIGURE 2.7: **DevR regulon.** **A)** *Left:* Histogram in blue represents the correlation among all the genes present in our compendium, whereas the histogram in green is based upon the correlations of the identified targets for which expression data is available within our compendium (605). Both show the same overall distribution. *Right:* Histogram in blue represents the correlation among all the genes present in our compendium, whereas the histogram in green is based upon the correlations of the 107 genes selected in the core set. Note there is a shift towards positive correlation values, pointing to a common regulatory influence over the selected genes. **B)** Number of genes identified in the DevR regulon compared to the number of targets identified through ChIP-seq experiments or the ones cited in literature [36, 98, 381]. **C)** Group assignment of the 622 targets identified by ChIP-seq for DevR. G0 contains genes for which non discernible expression pattern has been found. G1 correspond to the usually named DevR regulon, whereas G2-4 contains genes that show correlated expression patterns, although these patterns are not consistent with the previously described behaviour of DevR regulon. A detailed list of these genes is available in Additional file 12 and the output of the GO-enrichment analysis is shown in Additional file 16.

and are required for mycobactin-mediated iron acquisition [447] in *Mtb*.

Rv1734c belongs to the set of genes that had been previously reported as DevR regulated but for which our analysis shows that their expression pattern is not compatible with this assertion. Our analysis is based on transcript levels, so it could be that the regulation of Rv1734c happens at the post-transcriptional level. However we have found additional evidence, albeit indirect, of our prediction. Chauhan *et al.* [101] analysed the effect of mutations in the DevR binding motif and found that positions 5 (G), 7 (C) and 9 (A/T) are essential and a substitution in any of them dramatically reduces the binding affinity (see Additional file 15 for the DevR binding motif). In addition, our analysis shows that position 6 always contains an A. Motif analysis complements the results from the analysis of expression data and further supports our prediction that *Rv1734c* is not regulated by DevR, since its putative binding site contains a mutation (to G) at position 9.

Among the targets identified by ChIP-Seq four other groups emerge (see Figure 2.7C and Additional file 12) however none of these groups show significant up or down regulation in the conditions associated with DevR regulon induction. No clear common behaviour can be detected among the genes in group G0. These could be considered as false positives as a result of the textitdevRover expression performed prior to the ChIP-seq procedure [181]. However the genes within each of the other groups (G2, G3 and G4) show a consistent co-expression across the conditions in the compendium, therefore we believe that these hits should not be discarded as being caused by non-biological reasons, instead alternative explanations such as weaker binding to regulate overall or regulation through a transcription factor homologous to DevR should be further explored. In addition the genes in each group are functionally related (see Additional file 16), specifically, genes in G2 are mostly linked to translation and might be linked to the translation arrest observed during dormancy. Genes in G3 and G4 are functionally linked to 'metabolism', 'stress' and 'cell wall formation', which are significantly different in the non replicating state.

Zinc uptake regulator ZUR

As previously stated, the systematic analysis of genes linked to the different COG categories, showed that in the different co-expression networks the *rpmB2-rpmG1-rpsN2-rpsR2* operon (*Rv2058c-Rv2055c*) appeared forming a distinct cluster, separated from the rest of ribosomal protein coding genes (see Additional file 8). In addition, this operon also appears linked to the *ppe3-Rv0281* operon. Therefore, we concluded that these two operons should be regulated by a specific mechanism and respond to a specific type of perturbations. The location of these genes in the network of known regulatory interactions [431], pointed to the zinc uptake regulator ZuR (Rv2359) as the most likely regulator of both operons. Therefore we set forth to study the possible targets of this TF. Initially we selected from the network extracted from the literature, a list of regulatory targets of ZuR that had been verified either by more than one source or by an appreciable upregulation in a *zur* knock out mutant [312]. Once this core set was selected, we proceed to identify the subset of conditions where these genes are co-expressed. The method to construct the expression plots (see Materials and Methods) was used to select 23 conditions were the members of the core set were correlated (average correlation 0.76, see Figure2.8A); similarly we selected a core subset set of 35 conditions (average correlation of 0.65) and another subset of 5 conditions (average correlation 0.92). The correlation among the different putative members of the ZurR regulon (identified by motif elicitation and matching) was used as a signature to identify the other members of the regulon. The list of genes identified as belonging to this regulon is provided in Additional file 17 and the ZuR consensus binding motif is depicted in Figure 2.8A. The correlation analysis shows the importance of the biclustering approach to select only those conditions where no additional regulatory influences hinder the discovery of the members of the regulon. For example, if we analyse the correlation of *rpmB1* with the members of the core set across all conditions present in our compendium, we obtain no correlation (0.02). However when we compute the correlation between *rpmB1* and the genes assigned to the core set but only considering the previously selected subsets of conditions the correlation values raise to 0.63, 0.71 and 0.93 respectively, showing that, as previously reported, expression of *rpmB1* is indeed regulated by ZuR.

Our predictions closely match the list proposed by Maciag *et al.* [312], however some differences in the target assignment appear (Figure 2.8C). From the 34 previously assigned genes to the ZuR regulon, we can only confirm that 79% (27) of them show correlated expression patterns. In addition, among our predicted targets there are 10 genes for which no experimental evidence can be found in the literature. These predicted additional members of the regulon are: *Rv0223c*, *lpqR (Rv0838)*, *Rv1057*, *pe15 (Rv1386)*, *ppe20 (Rv1387)*, *Rv2617c*, *Rv2618*, *Rv2619c*, *Rv3018c* and *Rv3018b*. *Rv0232* and *lpqR* appear anti-correlated with the genes in the previously defined core set. In the case of *lpqR* this anti-correlation clearly increases when restricting the set of conditions. This together with the presence of ZuR binding motif in its upstream region, leads us to conclude that regulation of *lpqR* (and possibly of *Rv0232*) expression proceeds through a different mechanism than that of the rest of the genes in the regulon.

Additionally, it is striking to notice the clear upregulation of this set of genes in conditions where transcription was inhibited by addition of Rifapentine to the medium (conditions 207 and 208 from Additional file 1). This shows that, at least regarding these genes, the inhibition of transcription cause by Rifapentine has a similar effect as zinc limitation. However, a detailed explanation of this phenomenon is most likely unreachable from the analysis of the present dataset.

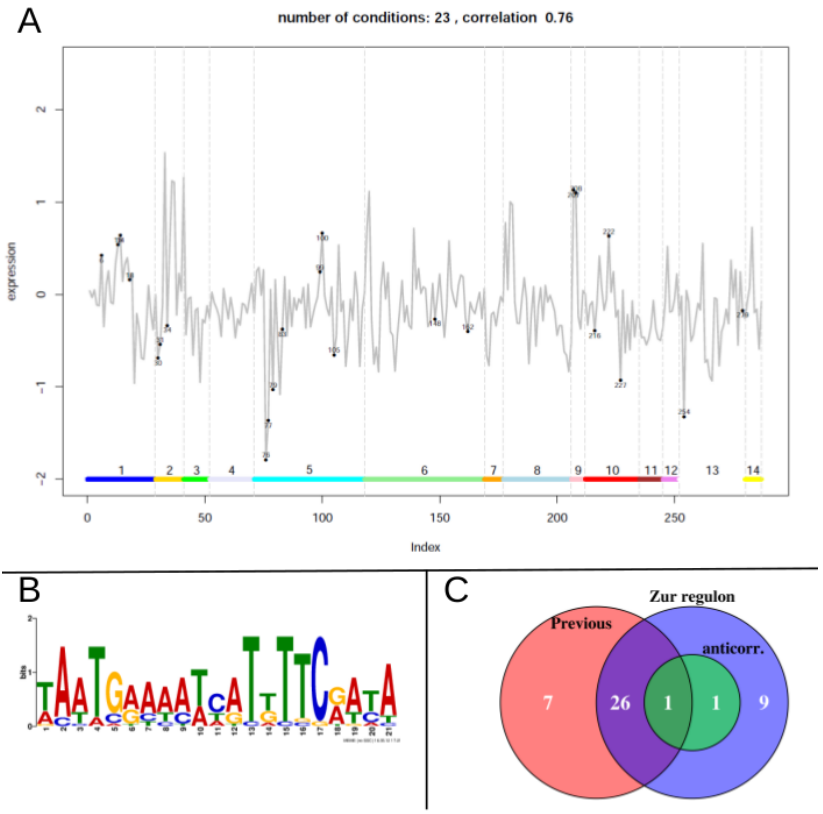


FIGURE 2.8: **ZuR regulon. A)** Bicluster formed by members of the ZuR as reported in literature [312]. The grey line represents the average expression levels of the members of ZuR regulon in the conditions in our compendium. The numbers identify the 23 conditions that have been included in the bicluster. The horizontal bar and the different regions indicated by numbers refer to the classification of the conditions as described in Materials and Methods. Notice the clear up-regulation of this set in conditions of type 9: Transcription inhibition, in particular these values correspond to experiments were Rifapentine was added to the medium. For clarity, expression values have been scaled, so that the mean value for each gene when all conditions are considered is zero. **B)** Identified ZuR binding motif. E-value 4.210^{-49}. **C)** Number of genes in the ZuR regulon (Additional file 17) compared to the ones previously identified [312], the set anticorr. contains *Rv0232* and *lpqR*, that show anticorrelation with the rest of the genes in the regulon.

Conclusion

We have shown that by integrating data from different sources and through the combined analysis of data, we are able to obtain insights into the biological system under investigation that go beyond the specific research questions of each experimental design. Our systems level approach has allowed us to analyse in depth publicly available data on *Mtb* and has enabled us to extract valuable new information from the existing datasets. The processes that we have studied (namely: adaptation to hypoxia, low zinc availability and DNA damage repair systems) are paramount in allowing *Mtb* to thrive within the hostile host environment. We generated comprehensive lists of genes involved in the response to such environmental conditions. These compendia summarize and substantially extend and modify the current knowledge. We believe that any further research on these adaptation mechanisms will make extensive use of this new knowledge and the hypotheses generated herewith.

We have developed a framework that allows the user to integrate information from gene or protein expression experiments, genome annotations and existing databases together with analysis tools. Most of the already existing databases and tools provide human user interfaces that only allow for querying one gene at a time and only provide a subset of the total available information. Within our framework this integration is done at once on a genome-scale, by using the gene co-expression networks. Instead of obtaining a network with the majority vote We have developed a framework that allows the user to integrate information from gene or protein expression experiments, genome annotations and existing databases together with analysis tools. Most of the already existing databases and tools provide human user interfaces that only allow for querying one gene at a time and only provide a subset of the total available information. Within our framework this integration is done at once on a genome-scale, by using the gene co-expression networks. Instead of obtaining a network with the majority vote.

Comparison with experimentally verified networks (*E. coli* and *S. cerevisiae*) and with *in silico* generated datasets shows that the co-expression networks generated using our pipeline preserve the desired modularity of transcriptional networks and the regulatory targets of a TF tend to appear in the networks as forming a closely interconnected module. Comparison with *E. coli* data, shows that in 97% of the cases the target genes of a TF form clusters in the network. Modularity was on average less preserved for *S. cerevisiae* networks (57% of the cases), due most likely to the increased complexity of regulatory interactions in eukaryotes. This poorer performance of the inference methods for the yeast dataset was also observed in the DREAM5 challenge were at most a 0.25 recall was obtained by any of the tested methods. On average the ZRC networks perform better that the mixed ones, but our analysis also showed that both types of networks contain complementary information, for example the targets of a yeast's YPR199C are only identified as forming a module in the mixed network (see Additional file 6).

We have established a protocol to assign regulatory interactions to binding sites identified through ChIP-seq experiments via the integration of expression data. We have used this approach to correctly identify the target genes of a given TF (DevR) as a response to an specific type of perturbation (dormancy inducing conditions) among the hundred of candidates from the experimental dataset. Within our framework biclusters can quickly be analysed, interactions between the biclusters can be identified in the overall networks and biclustering methods can easily be transformed into a tool for automatic detection of functionally related modules or underlying layers of regulation. Moreover, we have developed a method that allows to identify additional regulatory layers. The comparison of the networks generated by considering different subsets of conditions allowed us to distinguish various regulatory mechanisms for DNA damage response.

The analysis of ZuR regulon shows the potential of an integrative approach. In our compendium there were no data corresponding to experiments designed to analyse the effects of zinc limitation on Mtb. However, the analysis of the expression patterns of the genes in the different conditions and the analysis of their correlations, allowed us to select a set of conditions that would complement the bioinformatic analysis of the upstream sequences of the genes and would allow us to decided which of the regions similar to the motif actually represents a ZuR binding site and characterize the members of the regulon. We were able to compare our predictions with dedicated experiments performed with knock-out mutants and we found a good agreement between computational predictions and experimental data.

Our work extends the already existing knowledge and produces a comprehensive list of the members of the DevR regulon. Among the targets identified by ChIP-seq, we have uncovered three additional sets of genes that show a consistent expression pattern across the conditions in the compendium and are functionally related. Additional studies are required to further understand the regulation of these genes and their possible link to the non-replicative state. The discovery of new regulatory mechanisms involved in dormancy has the potential to deliver a new set of drug targets.

The automatic biclustering was the basis of our analysis of DNA repair systems and specifically LexA regulon. We were able to confirm our computational predictions through the comparison we literature data and recently performed ChIP-seq experiments. In addition, our work has produced a more specific binding motif for LexA through the identification of new members of its regulon. Additionally, the re-annotation of the identified new targets allowed us to identify a putative MR system in *Mtb*. The analysis of mutations and mechanisms to avoid them is of the uttermost importance for the further understanding of the evolution of antibiotic resistances and pathogenicity.

Previously existing data were used to verify our predictions on the regulatory mechanisms of DNA damage response. We have clarified some points previously in dispute, such as the lack of involvement of *sigG* in the response to DNA damage and the regulation of the alternate sigma factor *sigC* in these conditions. Correct identification of the sigma factor up-regulated upon DNA damage is key to understanding the systemic response of *Mtb* to this damage type. In addition, our work has provided further evidence on the mechanisms leading to cell cycle arrest upon DNA damage in *Mtb*.

In addition, we identified a new regulatory mechanism for ZuR, since the analysis of the upstream regions of its target genes and their expression patterns show that *Rv0232* and *lpqR* belong to its regulon, although their regulation must proceed through a different mechanism.

Here, we have presented the results obtained by applying our integrative approach to *Mtb* and whenever additional data was available, we have found good agreement between predictions and experiments. The basic underlying principle of this approach relies on the comparison of the networks obtained using different sources of information or methodologies. This approach can be readily extended to different organisms and to the comparison between different species, by using global identifiers together with a database of orthologous genes between species. This would allow to select a gene or group of genes in one organism and see how they are arranged in the networks corresponding to a different organisms. In addition, other types of data, such as synteny or evolutionary information, and protein structure and families could improve the evolutionary comparison of functional modules.

Funding

This work has been supported by European Union through the SysteMTb project (HEALTH-F4-2010-241587) and the MICROME project, EU Framework Program 7 Collaborative Project [222886–2] and by the Netherlands Consortium for Systems Biology through the "NCSB Stimulus 2013 project: data integratie" project.

Additional Files

Electronic supplementary material can be accessed at the on-line version of **Jesse CJ van Dam**, Peter J Schaap, Vitor AP Martins dos Santos, and Maria Suarez-Diez. "Integration of heterogeneous molecular networks to unravel gene-regulation in *M. tuberculosis*". In: *BMC Systems Biology* 8:111 2014.

Chapter 3

SyNDI: Synchronous Network Data Integration framework

Adapted from:

Erno Lindfors*, **Jesse C.J. van Dam***, Carolyn Ming Chi Lam, Niels Zondervan, Vitor A.P. Martins dos Santos, and Maria Suarez-Diez. "SyNDI: Synchronous Network Data Integration framework ". Accepted in *BMC Bioinformatics*

*Equal contributions

Abstract

Background: Systems biology takes a holistic approach by handling biomolecules and their interactions as big systems. Network based approach has emerged as a natural way to model these systems with the idea of representing biomolecules as nodes and their interactions as edges. Very often the input data come from various sorts of omics analyses. Those resulting networks sometimes describe a wide range of aspects, for example different experiment conditions, species, tissue types, stimulating factors, mutants, or simply distinct interaction features of the same network produced by different algorithms. For these scenarios, synchronous visualization of more than one distinct network is an excellent mean to explore all the relevant networks efficiently. In addition, complementary analysis methods are needed and they should work in a workflow manner in order to gain maximal biological insights.

Results: In order to address the aforementioned needs, we have developed a Synchronous Network Data Integration (SyNDI) framework. This framework contains SyncVis, a Cytoscape [443] application for user-friendly synchronous and simultaneous visualization of multiple biological networks, and it is seamlessly integrated with other bioinformatics tools via the Galaxy platform [65, 194, 198]. We demonstrated the functionality and usability of the framework with three biological examples - we analyzed the distinct connectivity of plasma metabolites in networks associated with high or low latent cardiovascular disease risk; deeper insights were obtained from a few similar inflammatory response pathways in *Staphylococcus aureus* infection common to human and mouse; and regulatory motifs which have not been reported associated with transcriptional adaptations of *Mycobacterium tuberculosis* were identified.

Conclusion: Our SyNDI framework couples synchronous network visualization seamlessly with additional bioinformatics tools. The user can easily tailor the framework for his/her needs by adding new tools and datasets to the Galaxy platform.

Background

Systems biology promotes a holistic approach in which biological elements such as molecules or reactions are no longer considered in isolation but as components of a bigger system such as a cell [270]. Within this framework, networks provide a natural way to describe associations and interconnections between system components. Network biology has emerged as one of the core sub-fields of systems biology in which nodes are biomolecules (e.g. proteins, genes, and metabolites) and edges represent interactions, associations and relationships between the biomolecules (e.g. chemical conversions, signal transduction steps, regulations, and co-expressions) [194, 270]. This approach is creating new inroads to solutions and applications in systems medicine [194, 270] and industrial biotechnology [369] among others.

The reconstructed networks are usually mined using a variety of querying methods [108, 335, 369]. In many cases, these methods aim at selection of sub-networks based on experimental evidence or on local topological properties (e.g. identification of network clusters) [383]. Computational analysis methods are in turn applied on selected sub-networks to understand related biological context. For example, Gene Ontology (GO) enrichment analysis can be performed to associate sets of genes or proteins with a specific biological process [28] or motif identification in upstream regions of selected genes [34] to identify gene regulators.

Biological network visualization has remained a highly non-trivial task and one of those currently open challenges related to the need of simultaneous network visualization to optimally and efficiently perform differential network analysis. Many alternative methods can be used to extract networks from the same datasets and the resulting networks have to be examined to generate a consensus network [319]. Different network representations are needed to convey different layers of information pertaining the same system (e.g. metabolic networks, protein-protein interactions networks, gene regulation networks), however these information layers are not independent and all of them have to be considered as a whole in order to describe how the overall system behaves. Moreover, different networks might arise even when considering similar biological processes under different conditions (e.g. healthy versus disease states) [236].

As a result of this multiplicity in the nature of networks and the subsequent integration need, many advanced graph-based methods have been developed for comparing networks [236]. Some of them produce local measures for individual nodes (e.g. node degrees, clustering coefficients) and these are compared on a node basis across different networks. This can, in turn, help clarifying the biological significance of a highly connected node, or hub. Other methods give global measures for the network as a whole, for instance distributions and average values for node degree and clustering coefficients, and network diameter [404]. A researcher is needed to interactively inspect these results to achieve proper analysis and interpretation.

As stated, network analysis requires the use of complementary analysis methods. In today's omics era it has become utmost important that these data analyses can be performed through the use of consistent workflows, where results can be stored for further analysis and findings can be reliably reproduced. In addition, these workflows have to be integrated with network visualizations, so that it is possible to easily switch from network interpretation to subsequent bioinformatics data analysis and vice versa. Galaxy is a user-friendly web-based platform that has been developed to address these needs [65, 194, 198].

Here we present SyNDI, a Synchronous Network Data Integration framework for synchronous visualization of multiple biological networks that addresses the above mentioned challenges. Specifically, the SyNDI framework endows Cytoscape [443] with the capability to show multiple networks in a synchronous way that preserves the identity between nodes appearing in multiple networks, thus enabling visually inspecting differences in their local connections. SyNDI also provides the possibility to perform data analysis directly from the network visualization (without complicated file handlings) using Galaxy and vice versa - the analysis results from Galaxy can be directly exported to the network visualization.

Here we demonstrated the functionality and usability of SyNDI with three biological examples. First, we illustrated how it can be used to assist analysis of metabolite association networks related to high and low latent cardiovascular risk respectively by simultaneously visualizing those networks. In our second example, we analyzed a few common response pathways between human and mice in *SStaphylococcus aureus* infection to gain further biological insights. Finally, we demonstrated how SyNDI connects network visualization with Galaxy's data analysis tools and specifically we analysed type VII secretion system, ESX-1, in the human pathogen *Mycobacterium tuberculosis*; this study represents a follow-up on an earlier analysis of key regulatory events associated with pathogenesis and survival within the host, see [123].

Implementation

The overall architecture of our framework is presented in Figure 3.1. It is composed of two layers: SyncVis is a Cytospape app that allows the user to visualize multiple biological networks exploiting the Cytoscape core and network analysis layer which uses Galaxy [65, 194, 198] for central core of analysis. In the next sub-sections we describe these layers in technical detail.

Network Visualization

We have developed a Cytoscape app called SyncVis (Synchronous Visualizer) for network visualization. Also we use Cytoscape core for some of this functionality. In the next sub-sections we describe the technical implementation

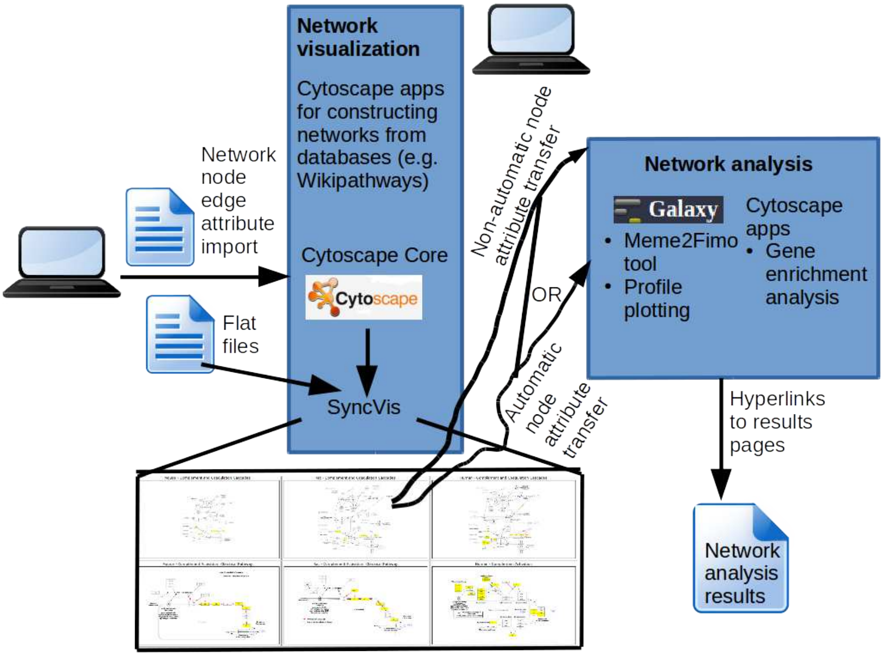

Network visualization
Cytoscape apps for constructing networks from databases (e.g. Wikipathways)
Cytoscape Core
Cytoscape
SyncVis
Network node edge attribute import
Flat files
Non-automatic node attribute transfer
OR
Automatic node attribute transfer
Network analysis
Galaxy
• Meme2Fimo tool
• Profile plotting
Cytoscape apps
• Gene enrichment analysis
Hyperlinks to results pages
Network analysis results

FIGURE 3.1: **Technical architecture of a workflow system.** It comprises of layers for network visualization and analysis; synchronous network visualization on a SyncVis Cytoscape app and network analysis on Galaxy or another external tool. The user can transfer node attributes from SyncVis to a network analysis to automatically or non-automatically.

in detail. In the first sub-section we present a few options the user can apply for constructing networks as a pre-step before starting to use the SyNDI framework. Then we describe three other sub-functionalities: Network import, synchronous visualization and node attribute export.

Pre-step - Network construction

In order to visualize networks on the SyNDI network, the user needs to construct networks. We would like to emphasize this procedure is not part of SyNDI framework. However we feel this procedure deserves its own subsection since it is a necessary pre-step - the user needs to have sufficient knowledge about network construction in order to use the SyNDI framework.

She can use a top-down approach to generate networks from experimental data using existing reconstruction algorithms [319]. In most cases it is pragmatic to implement these algorithms as separate applications for example in the R environment.

Alternatively the user can use a bottom-up approach by constructing networks from available biological databases (e.g. signaling pathway databases, metabolic pathway databases, protein-protein interaction databases). Most of biological pathways have some networks directly available on their web sites - for example Wikipathways database [283, 319] has hundreds of pathways available (`http://www.wikipathways.org`). In addition, these pathways usually have Application Programming Interface (API) available that support high-level programming languages (e.g. Java). The user can use these APIs to implement an application customized for her purpose. Some of these databases have been integrated in common bioinformatics tools - for example Wikipathways database has a Cytoscape app (`http://apps.cytoscape.org/apps/wikipathways`) that the user can use to retrieve pathways based on various search parameters directly on Cytoscape.

Network import

Cytoscape core supports most of the base network representation formats like Simple Interaction Format (SIF), eXtensible Graph Markup and Modeling Language (XGMML) and Systems Biology Markup Language (SBML). Most of the networks construction tools and methods covered in the previous subsection can generate networks in some of these formats. The user can therefore import networks to Cytoscape Core for example by using "ctrl + L" shortcut key or "File -> Import -> Network" menu. In some case the user may have additional parameters for nodes or edges in separate files. This can be the case for example if she has used a separate tool to calculate log2 fold changes and statistical metrics like p-values from transcriptomics data. Technically this happens by using the "File -> Import -> Table" menu on Cytoscape core.

Alternatively the user can use a specific Cytoscape app like the Wikipathways Cytoscape app mentioned in the previous sub-section to construct networks directly on Cytoscape.

Synchronous visualization

The concept of synchronous network visualization is illustrated on the bottom of Figure 3.1. Typically the user goes though the following pipeline when using this feature.

- The user has a specific node of interest (e.g. an individual gene) or a group of nodes (e.g. genes involved in a specific biological process).
- The user search the node(s) on one network (e.g. on an organism specific pathway).

- The corresponding node(s) are automatically highlighted on another network (e.g. on a similar pathway from another organism). The user can thus easily look into the differences in local connections of the nodes between the networks.

The same pipeline can be applied to a synchronous visualization of any other networks (e.g. networks from different medical conditions, networks produced by different network construction algorithms).

We have implemented a Cytoscape app called SyncVis (Synchronous Visualizer) for this functionality by using Cytoscape Java API package (`http://chianti.ucsd.edu/cytoscape-3.5.1/API/`). We map the node selections via Cytoscape's "shared name" attribute which means that node identifies (e.g. gene names) have to be stored in this attribute. Next we will present simplified code snippets demonstrating how these mappings are implemented on Java programming level.

```
// First we retrieve selected nodes from Cytocape's "selected"
// attribute: selCyNet the network on which the user selects
// the nodes.
List<CyNode> selNodes
   = CyTableUtil.getNodesInState(selCyNet,"selected",true);

// Then we store the "shared name" attributes of the selected
// nodes in a hash set:
HashSet<String> selSharedNames = new HashSet<String>();
for (CyNode node : selNodes) {
  String sharedName = cyNodeTable.getRow(node.getSUID())
                        .get("shared name",String.class);
  selSharedNames.add(sharedName);
}

// Then we select the nodes of the other networks based on
// their presence in selSharedNames: allNets is a list that
// contains all networks that are imported in Cytoscape
for (CyNetwork cyNet : allNets) {
  CyTable cyNodeTable = selCyNet.getDefaultNodeTable();
  for (CyNode node : selCyNet.getNodeList()) {
    CyRow row = cyNodeTable.getRow(node.getSUID());
    String sharedName = row.get("shared name",String.class);
    row.set("selected", selSharedNames.contains(sharedName));
  }
}
```

In addition the user can upload his/her own mapping file (e.g. homologs between two species). We have explained this procedure in the user manual.

Node attribute export

SyncVis needs functionalities to export node attribute data for smooth communication with network analysis. Cytoscape's "shared name" attribute is used to for this connection and it is accessed in the same way on Java programming level as in synchronous visualization as described in the previous sub-section.

As indicated in Figure 3.1, SyncVis contains two alternative options for the export:

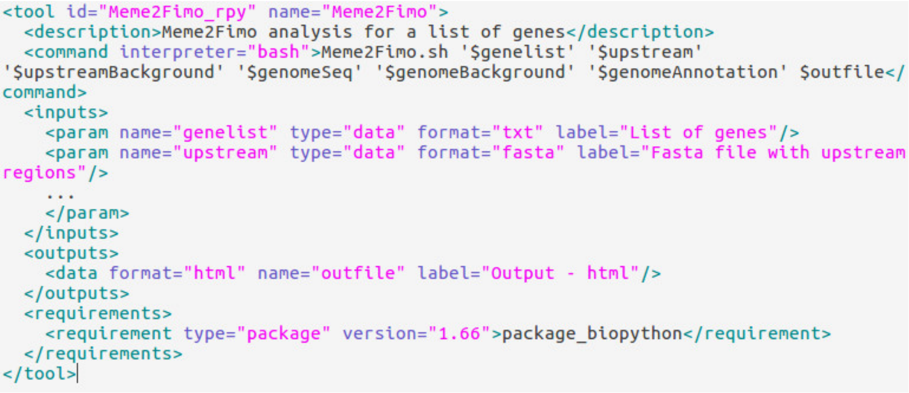
- *Automatic export* In this option the node data transfer from SyncVis to the Galaxy platform is automated; SyncVis communicates automatically with Galaxy and the user does not need do any manual operation. SyncVis contains two buttons to this operation for each network analysis: one button that creates a flat file when the user clicks on it and another button that sends the request the Galaxy platform when the user clicks on it and has possibly given additional parameters for network analysis. Technically this is implemented so that first SyncVis creates a flat file that contains the "shared name" attributes of the selected nodes. Then it calls a python script from Java code that uses a BioBlend API [458] to send the flat file to the Galaxy platform as an input of network analysis.

- *Non-automatic export* In this option user interventions is needed for the node data transfer from SynVis to the Galaxy platform (or another network analysis tool such as a (Biological Networks Gene Ontology tool

(BiNGO) Cytocape app [315]). First a flat file is created. This is done manually and using a button on SyncVis to copy-paste the “shared name” attributes of the selected nodes to the flat file. Alternatively the user can click another button on SyncVis to save the attributes to a flat file. The flat files can then be imported into Galaxy or another network analysis tool.

SyncVis has these two alternative options in order to find a balance between automated communication and software development. The automated export option is very user-friendly but some technical work is needed to implement it on SyncVis for a network analysis tool. For time being this export is therefore implemented only for a few network analysis tool. The non-automatic export is not so user friendly but this is does not require any extra work from the software developer, so the user can use it immediately if she wants use a network analysis tool for which the automatic export is not implemented.

Figure 3.2 illustrates the content of an Extensible Markup Language (XML) file defining a connection between SyncVis and a tool running on the Galaxy platform. The command element defines how the Galaxy platform executes the tool using the input files listed in the input element. The output element defines the format of the response. Algorithms can be implemented by any programming language that Galaxy supports (e.g. R, Python, bash). More details about the content of this file can be found at the tool configuration page at the wiki page of the Galaxy project (`https://docs.galaxyproject.org/en/latest/dev/schema.html`).



```
<tool id="Meme2Fimo_rpy" name="Meme2Fimo">
  <description>Meme2Fimo analysis for a list of genes</description>
  <command interpreter="bash">Meme2Fimo.sh '$genelist' '$upstream'
'$upstreamBackground' '$genomeSeq' '$genomeBackground' '$genomeAnnotation' $outfile</
command>
  <inputs>
    <param name="genelist" type="data" format="txt" label="List of genes"/>
    <param name="upstream" type="data" format="fasta" label="Fasta file with upstream
regions"/>
    ...
    </param>
  </inputs>
  <outputs>
    <data format="html" name="outfile" label="Output - html"/>
  </outputs>
  <requirements>
    <requirement type="package" version="1.66">package_biopython</requirement>
  </requirements>
</tool>
```

FIGURE 3.2: **Content of an XML file that defines a Galaxy tool.** This file contains a brief description of the tool, a command for running the tool, and the input and output parameters of the tool.

SyncVis needs an API key for the connection with the Galaxy platform. Our user manual contains detailed instructions for configuring this key.

Network Analysis

The purpose of network analysis is to gain our understanding of the underlying biology behind a visualized network. The user can select sets of genes for further investigation on the visualized network. She can for example perform a Gene ontology (GO) enrichment analysis to see in what biological processes the genes are over-represented, or plot the gene expression profiles or search shared sequence motifs between the genes.

We use Galaxy as a central platform for running these analyses since it is a widely used platform for running bioinformatics analysis requiring no programming skills from end users. When the Galaxy platform has completed analysis, it reports the results on Hyper Text Markup Language (HTML) pages to which SyncVis displays links on pop-up windows.

The SyNDI framework is not restricted to Galaxy as it can easily interoperate with other available analysis tools and Cytoscape applications, such as BiNGO [315].

Meme2Fimo tool

We have implemented a tool called Meme2Fimo in the Galaxy server for upstream sequence analysis. Meme2Fimo integrates tools for motif identification (MEME [34]) and motif search (FIMO [205]).

From a user given gene selection, MEME is used to identify up to 5 possible motifs in the upstream regions of the selected genes, which are automatically retrieved from a GenBank file. MEME is executed with the "-dna -revcomp -nmotifs 5 -mod zoops -evt 1000" parameter string. MEME generates a list of found motifs and for each motif it returns an ordered list of scores for the selected input genes. The score indicates how well the motif fits to the identified upstream region. These motifs and associated ordered list are collected and stored.

For each motif identified by MEME, FIMO is executed to locate any other occurrences within the complete genome. FIMO is executed with the "–bgfile <genome background>" option. The genome background is generated from the complete genome sequence using the "fasta-get-markov" command, with an order value of 3. FIMO returns a list of occurences with an associated location, p-value and q-value. This list is ordered by p-values. All occurrences that occur within a known gene are rejected. For each remaining occurrence, Meme2Fimo searches for the gene downstream and upstream if present. If for the given downstream gene already another occurrence is found, then it is rejected and the hit count of the already found occurence is increased by one. If the downstream gene is present within the stored list captured from the MEME output, the index within that list is added to final output of Meme2Fimo. Otherwise a -1 is added. So Meme2Fimo will add for each motif result generated by MEME to an additional table, which contains a row for each accepted occurrence found in the genome of that motif: a downstream gene identifier, a sequence associated to the occurrence, an index of the downstream gene within the initial MEME result, an index of the upstream gene within the initial MEME result, a p-value, a q-value, a hit count and relative position to the downstream gene.

Based on the index values one can identify other genes that are regulated by the same regulator. If in the top hits within the list some occurrences and associated genes are found, which are not within the selected set of genes (indicated with a -1) one can add these genes to the input and rerun Meme2Fimo. If one keeps repeating this process, in some cases (e.g. in a motif related to the DosR regulator presented in Results and Discussion) the indexes in the list will converge to a list without any -1 values in between.

Results and Discussion

Probabilistic networks of blood metabolites associated to latent cardiovascular risk

Comparison of networks extracted under different clinical conditions, such as health and disease, might help uncover key mechanisms of disease physiology, especially in conditions whose outcome is presumably affected by a multitude of risk factors. Cardiovascular diseases (CVD), one of the leading causes of death in western countries, are associated to risk factors of metabolic origin, however the complex nature of CVD has prevented a complete mechanistic understanding of these risk factors and their associations.

In a previous study [425], a global analysis was performed on the association networks between a panel of metabolites quantified using Nuclear Magnetic Resonance (NMR) from plasma samples from healthy individuals. Metabolites' association networks were defined for individuals with low CVD risk and for those presenting latent CVD risk. Briefly, an array of 29 metabolites identified and quantified in the plasma of 864 healthy blood donors of both genders was considered [59]. Clinical data and traits: concentrations of high and low density lipoproteins (HDL and LDL respectively), total cholesterol, triglycerides, glycaemia and Framingham score, were used to split the cohort according to latent CVD risk levels: low, medium or high. Metabolite networks associated to high and low CVD latent risk were extracted using the Probabilistic Context Likelihood of Relatedness based on Correlation (PCLRC) algorithm [425].

Figure 3.3 represents the associations linked to either high (panels A and C) or low (panels B and D) latent CVD. Topological indices for each node, such as clustering coefficient and degree are represented by node color and size respectively. Using a common layout for both networks eases the comparison, as nodes occupy the same relative position in both networks (compare panels A and B of Figure 3.3). However, a dedicated layout for each of them, (Figure 3.3 C and D) eases the identification of the key local connections. These network representations emphasize , for instance, the prominent location of very-low-density lipoprotein (VLDL) in the high latent risk network (Figure 3.3 C) or the two connected components in the low CVD risk network (Figure 3.3 D) that highlights the association between acetate and the amino acids serine, histidine, phenylalanine, glutamine and alanine. In the high latent risk network these latter associations are disrupted and glucose appears associated to amino acids, which are known mediators of glucose metabolism, insulin secretion, and insulin sensitivity [336].

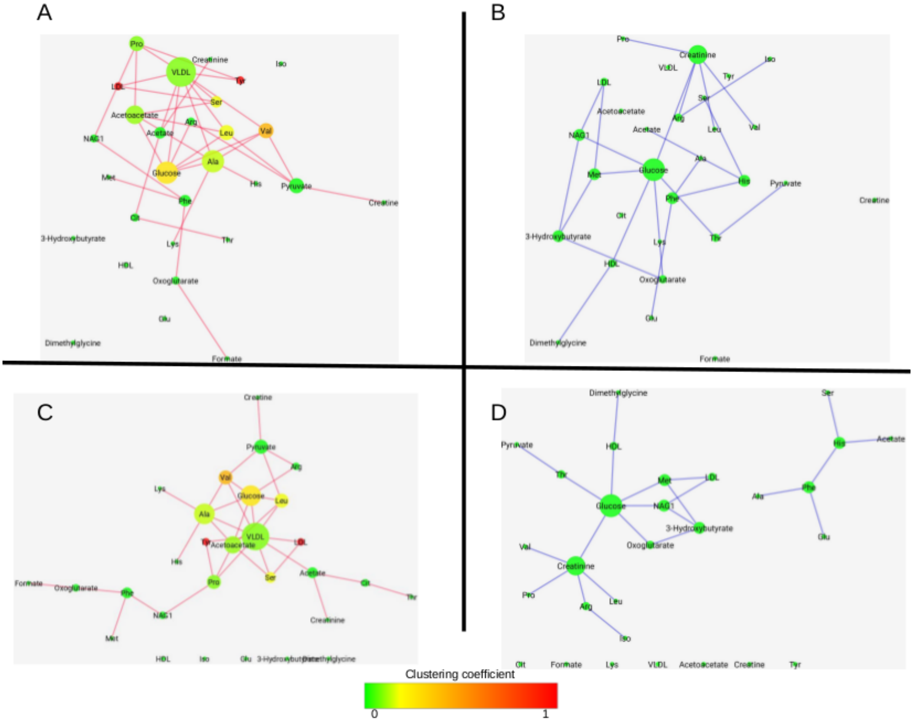


FIGURE 3.3: **Association networks of blood metabolites.** Nodes represent metabolites. Node size is proportional to node degree and node color is linked to clustering coefficient. A) and C): Associations found exclusively in subjects with high latent CVD risk (red edges). B) and D): Associations found exclusively in subjects with low latent CVD risk (blue edges). Networks in A and B have the same node location. Networks C and D have been obtained using force directed layout in each of them.

However, these networks pe se are not enough for a smooth local view switch network. SyncVis tackles this challenge by transferring node selections between networks automatically.

Synchronous visualization of differentially expressed genes under *S. aureus* infection on human and mouse signaling pathways

In order to demonstrate SyNDI's functionality for synchronous network visualization of networks across different species, we visualized Differentially Expressed (DE) genes in the context of *S. aureus* infection in human and mouse. We thus aimed to gain deeper insights among dysregulated pathways shared by these two species during *S. aureus* infection. Banchereau et al. performed whole transcriptomics analysis on *S. aureus* infected patients and healthy people (99 and 44 samples respectively) [38]. This data set comprises 24 371 transcripts. DE genes were identified (False Discovery Rate (FDR) < 0.01 and log2 fold change > 0.7). Brady et al. studied protective mechanisms in mice to *S. aureus* Skin and Soft Tissue Infection (SSTI) [75]. They used an SSTI mouse model to study local (=infected vs non-infected ears) and systemic (=challenged vs naïve mice) responses to infection at one, four and seven days after the start of infection. RNA sequencing (RNA-seq) was used and DE genes were defined as those with a log2 fold change of 1 or higher. We selected the local response at four days for our study as this time point gave the most significant overlap with WikiPathways.

We retrieved all human and mouse signaling pathways from the WikiPathways database [266, 283]. 25 pathways with at least 4 DE genes in both human and mouse were selected (see table in Additional file 1).

Three pathways from this table were visualized using SyNDI to illustrate how its synchronous network visualization functionality provides an easy and effective approach to compare pathways between human and mouse. Detailed step by step instructions to run these examples are provided in Additional file 2. All needed scripts and data files are provided in Additional file 3.

Complement and Coagulation Cascades

As indicated in Additional file 1, this pathway (Figure 3.4) has one of the largest number of DE genes among those already reported in the literature to be differentially regulated in both human and mouse blood samples under various injury or bacterial infection conditions (including *S. aureus* infection). Nearly all DE genes in this pathway were up-regulated. The complement system and coagulation system are main columns of innate immunity and hemostasis respectively [16], so their up-regulation in human and mouse indicated an attempt of the hosts to fight against injuries or infections and to recover from damage. Among those 12 DE genes in this pathway found in human and mouse datasets, only 3 genes (F5, C1QB, and C3AR1) are homologs and they appear significantly up-regulated in both cases. Using SyNDI's synchronous visualization, one can immediately identify that C1QB and C3AR1 belong to the classical pathway of the complement cascade, but F5 is among several other up-regulated genes in the coagulation cascade. C1QB is a subcomponent subunit of C1Q. Deficiency of C1q has been reported to be associated with recurrent infections among Inuit people [323]. Literature studies about C3AR1 and bacterial infection are very limited. Antunes and Kassiotis [19] studied influenza A virus infection-induced pathology in lymphocyte-deficient mice. C3ar1 in cells of the monocyte/macrophage lineage was one of the most highly induced gene transcripts, suggesting a role of C3ar1 in infection. F5 is a central regulator of hemostasis. In mice, reduction of F5 in blood plasma or platelet caused higher mortality upon Group A *Streptococcus* infection, highlighting the importance of F5 pool in host defense [479]. Overall, this visualization feature has facilitated quick identification of common regulation trends in parts of the complement and coagulation cascades between human and mouse. It can also speed up comparison of DE genes which are different between the two species in this pathway for potential further investigation.

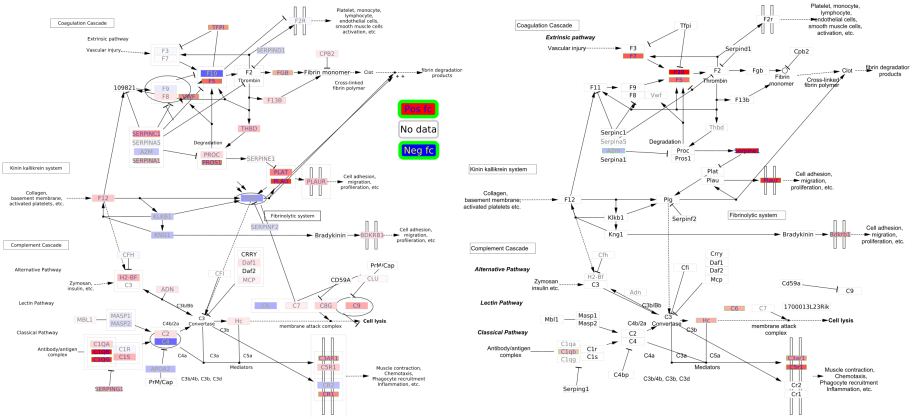


FIGURE 3.4: **DE genes on "Complement and Coagulation Cascades" pathway upon *S. aureus* infection, human pathway on the left part and mouse on the right.** Node color has been mapped to log2 fold change; red/blue denoting positive and negative values respectively (see legend). White color is used for nodes (genes or metabolites) for which either no data was available or changes were not deemed significant. The human pathway contains 169 nodes and 100 edges and the mouse pathway 148 nodes and 86 edges. Additional file 7 contains a Complement_and_Coagulation_Cascades_human_mouse.cys file which can be opened on Cytoscape to view these pathways with better resolution.

Wnt Signaling Pathway and Pluripotency

The Wnt signaling pathway has been reported in several studies as commonly regulated in human and mouse [7, 484]. Wnt signaling are responsible for cell differentiation, development, and tissue homeostasis etc. [222, 386]. A direct evidence for the relevance of Wnt5A in severe systemic inflammation is supported by the finding of higher Wnt5A levels in patients with sepsis than in healthy individuals [386]. Although all those DE genes in this pathway are different in human and mouse, from Figure 3.5 we can easily identify that a few genes belonging to frizzled ligands and some of the beta-catenin target genes in the nucleus are differentially expressed in both mouse and human. It is expectable that differences between species would result in different genes being regulated in similar pathways in human and mouse. Those commonly regulated sub-networks of the Wnt signaling pathway and pluripotency network as shown by the synchronous visualization are tentative leads for further investigation of common signaling mechanisms in human and mouse upon *S. aureus* infection.

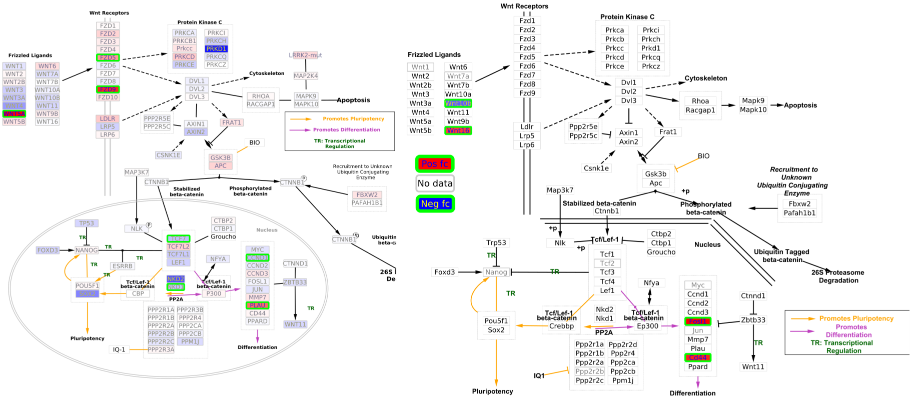


FIGURE 3.5: **DE genes on "Wnt Signaling Pathway and Pluripotency" pathway upon *S. aureus* infection, human pathway on the left part and mouse on the right.** See legend in figure 3.4 for additional information on coloring scheme. The human pathway contains 174 nodes and 55 edges and the mouse pathway 175 nodes and 54 edges. Additional file 7 contains a Wnt_Signaling_Pathway_and_Pluripotency_human_mouse.cys file which can be opened on Cytoscape to view these pathways with better resolution.

Insulin Signaling

The insulin signaling pathway contains 8 DE genes in human and in mouse, of which only SOCS3 is shared between the two species (Figure 3.6). All these DE genes were up-regulated in both species. Georgel et al. reported TLR2 affected the outcome of mouse skin infection by bacteria [192]. In a study of gut microbiota of type 2 diabetes and obesity subjects, it was observed that TLR2 and inflammatory pathways were activated in obese individuals and insulin signaling was impaired relative to lean individuals [93]. Although the involvement of insulin signaling in diabetes is well-known, the potential role of this pathway in bacterial infections is rarely studied in the literature. Mele and Madrenas [333] studied literature evidence of infections by *S. aureus* and suggested TLR2 signals can differentially induce SOCS1 and SOCS3. In Figure 3.6, both Socs1 and Socs3, belonging to modulators of insulin action, were significantly up-regulated in mouse. Further investigation is necessary to verify the potential relationship between *S. aureus* infection and insulin signaling pathway, but the network visualization approach has provided a convenient method to identify pathway candidates that appear to share unknown connections.

Identification of binding motifs associated to DosR in *M. tuberculosis*

A pipeline for the reconstruction of gene co-expression networks from a compendium of expression data was described in [123] to where we refer the reader for additional details. This pipeline is highly customizable and its default values correspond to the following brief description. From a gene expression compendium, similarity between gene expression profiles is scored using Pearson's correlation for each gene pair. The significance of the similarity is scored using an estimate for the null model based on the rest of the similarity scores obtained for the members of the pair evaluated independently [161]. A generalization of the data processing inequality is iteratively applied to prune possible spurious associations from the network [320]. Stand-alone scripts implementing this pipeline can be retrieved from Additional file 3 of [123].

We have used the Meme2Fimo tool to investigate transcriptional regulation of *M. tuberculosis*, the aetiological agent of tuberculosis. Specifically we investigated the role and regulation of ESX-1 associated genes *espA*, *C* and *D* and the role of DosR in regulating these genes. ESX-1 is a type VII secretion system required for the secretion of virulence proteins such as EsxA (ESAT-6) and EsxB (CFP-10). These are involved in immune modulation and phagosome escape [1, 452, 546]. EspACD is required for EsxA-EsxB secretion and pore formation [104, 184]. Multiple regulators such as PhoP, EspR, MprA, CRP are involved in modulation of ESX-1 and its secreted factors [256]. The transcription factor DosR (DevR) mediates the hypoxic response of *M. tuberculosis* and triggers the onset of dormancy which enables long term survival of the bacteria within the lung granulomas of the human host [381]. DosR regulon is essential for persistence and pathogenesis of *M. tuberculosis* [332]. ChIP-seq experiments initially identified over 600 gene targets for DosR [181] and its binding motif is shown on Figure 3.7 [101]. Integration of heterogeneous molecular networks with this data led to the identification of five groups of genes with distinct expression profiles among this initial set [123].

Here we used SyNDI framework to further investigate additional regulatory motifs related to ESX-1 systems by simultaneous exploration of the *CLR*,

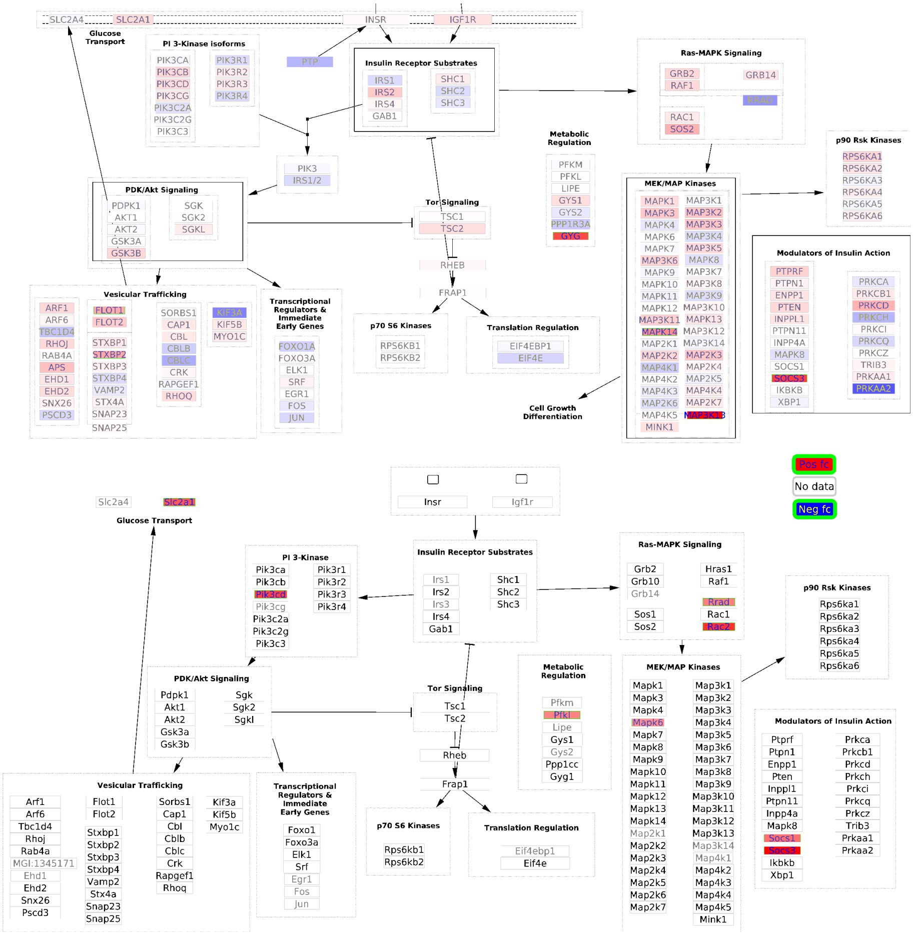


FIGURE 3.6: **DE genes on "Insulin Signaling" pathway upon *S. aureus* infection, human pathway on the top part and mouse on the bottom.** See legend in Figure 3.4 for additional information on coloring scheme. The human pathway contains 226 nodes and 25 edges and the mouse pathway 195 nodes and 15 edges. Additional file 7 contains an Insulin_Signaling_human_mouse.cys file which can be opened on Cytoscape to view these pathways with better resolution.

STRING.db fusion, *STRING.db neighbourhood*, *operon* and *BLAST* based homology (bbh) networks presented in [123], to where we refer the reader for additional information on these networks. Technical details are provided in Additional files 4 and 5.

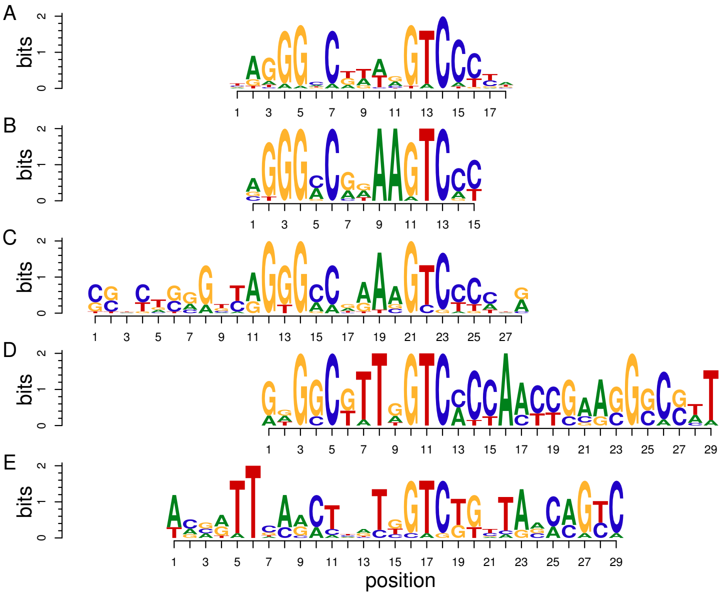


FIGURE 3.7: **Comparison of DosR and ESX-1 related motifs.** A) DosR motif as reported in [101] B) Exploration path 3 motif. C) Exploration path 2 motif 2. D) Exploration path 1 motif. E) Exploration path 2 motif 1.

Exploration path 1: ESX-1 associated genes espA,C and D

Initially, ESX-1 related genes, *espACD*, and other closely positioned genes in the CLR network were selected. The gene selection was transferred to the fusion network and three additional genes were identified in their neighbourhood. This selection was further enlarged with genes in their neighbourhood previously reported in the DosR regulon [123]. Transferring the selection to the bbh network led to the identification of three pairs of homologous genes. In each pair one gene belongs to the ESX-1 related gene set whereas the other one is in the DosR regulon (see Table 3.1). In the fusion network genes in these homology pairs within the DosR regulon appear as a densely connected cluster, together with Rv0080 and TB31.7. TB31.7 is a universal stress protein family protein responding to stress signals and has been shown to be involved in growth arrest during latent infection.

To further investigate the role of TB31.7 a new selection was made in the *bbh* network by adding six TB31.7 homologs, five of which are in the DosR regulon. Meme2Fimo was iteratively used to explore upstream sequences of these genes. Finally, a conserved motif similar to the one reported for DosR

TABLE 3.1: **Hypothetical homologous complexes.** Pairs of homolog genes in the ESX-1 and DosR related clusters of two hypothetical homologous complexes. *low similarity (E-value 3e-09 < network visualization threshold).

ESX-1 cluster related	**ESX-1 cluster related**
Rv0569	*Rv2302*
Rv2632c	*Rv1738*
*Rv2406c**	*Rv2626c**
	Rv0080
	TB31.7

was identified (Figure 3.7). However, some distinct features appear showing that regulation of ESX-1 related genes *espACD* is complex, integrating signals from hypoxia via DosR but also possibly increased cell stress signals via TB31.7 homologs.

Exploration path 2: TB31.7 and its homologs

To further investigate the *TB31.7* gene and its homologs, we selected them and neighbouring genes within the *neighbourhood* network. Upstream regulatory regions analysis lead to the description of another motif (Figure 3.7). A subset of genes (*Rv2621c*, *Rv2622*), coding for a possible transcriptional regulator and methyltransferase, with this motif in their upstream regions appear in the CLR network with a cluster of genes related to mycolic acid synthesis. The ratio of free and bound mycolic acids is known to change under hypoxia and cell wall stress [181].

We further investigated the DosR regulation of Universal Stress Protein (USP) homologs to TB31.7 and its relation to ESX-1. We described another motif in Figure 3.7.

Exploration path 3, likely sigE binding motif

We explored the DosR regulon to identify elements with additional regulatory influences. USPs homologs to TB31.7 with the DosR regulon and genes in the same operons were selected. Transferring the selection to the gene *neighborhood* network showed the relationship between these two related groups and suggested some genes to be further included in the selection. Yet another motif (Figure 3.7) was described in the upstream regions of these genes.

This motif is similar to the binding motif of the AlgU sigma factor from P. aeruginosa which is homologous to SigE in *M. tuberculosis* [166]. SigE and SigH together with MprAB function to detect and protect against cell stress such as misfolded proteins, heat shock, acidic pH, exposure to detergent, and oxidative stress. These conditions are associated with failed immune modulation which is related to the DosR regulated dormancy regulon [44, 166, 523]. Moreover, Rv0080, which is also in the DosR regulon, has been reported as a regulatory hub of the hypoxia response regulated by MprA [77, 181] .The identified binding motif shows similarity to the motifs detected upstream of genes experimentally shown to be regulated by SigE and SigH regulated genes [469].

Motif comparison

Figure 3.7 shows five related binding motifs. The location of these motifs is shown in Figure 3.8 and Additional file 5. The groups of genes controlled by this motifs are shared as shown in 3.9. Inspection of the locations of the motifs shows their overlaps in the upstream regions of the various shared genes of motifs B, C and D, which indicates that the shifted motifs might still be functional. The general DosR motif GGGNCNNNNGNCCC is palindromic, whereas motif B GGGNCNN**AA**G**T**C has a unique element, which is not palindromic. Both SigE and DosR are related to the modulation of process directly related to growth within human macrophages, the similarity between this motif and the AlgU motif in P. aeruginosa led us to hypothesize that DosR and SigE can bind to the same regions. Furthermore motif D GGGNCN**TT**NG**T**C also has a unique element, NAA in motif B is replaced by TTN.

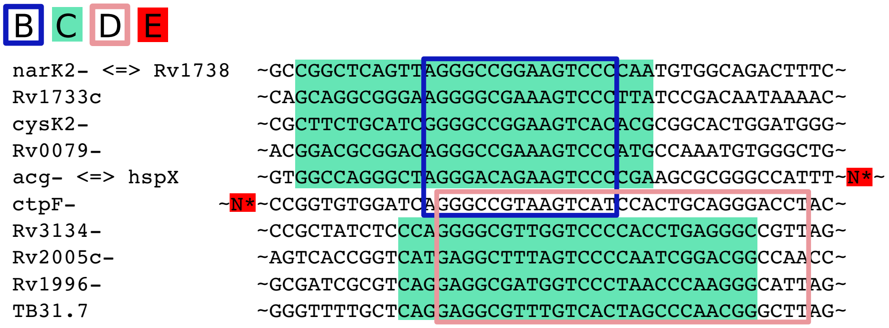

B
C
D
E
narK2- <=> Rv1738 ~GCCGGCTCAGTTAGGGCCGGAAGTCCCCAATGTGGCAGACTTTC~
Rv1733c ~CAGCAGGCGGGAAGGGGCGAAAGTCCCTTATCCGACAATAAAAC~
cysK2- ~CGCTTCTGCATCGGGGCCGGAAGTCACACGCGGCACTGGATGGG~
Rv0079- ~ACGGACGCGGACAGGGCCGAAAGTCCCATGCCAAATGTGGGCTG~
acg- <=> hspX ~GTGGCCAGGGCTAGGGACAGAAGTCCCCGAAGCGCGGGCCATTT~N*~
ctpF- ~N*~CCGGTGTGGATCAGGGCCGTAAGTCATCCACTGCAGGGACCTAC~
Rv3134- ~CCGCTATCTCCCAGGGGCGTTGGTCCCCACCTGAGGGCCGTTAG~
Rv2005c- ~AGTCACCGGTCATGAGGCTTTAGTCCCCAATCGGACGGCCAACC~
Rv1996- ~GCGATCGCGTCAGGAGGCGATGGTCCCTAACCCAAGGGCATTAG~
TB31.7 ~GGGTTTTGCTCAGGAGGCGTTTGTCACTAGCCCAACGGGCTTAG~

FIGURE 3.8: **Shifted motif alignment.** Marked region denotes the region containing the sequence to which the motif matches. The regions marked for the motif D regions are shifted. See Figure 3.7 for the legend.

The palindromic motif E lacks the characteristic GGGNCNNNNGNCCC pattern describing the general DosR binding motif. Only the GTC is conserved in comparison to the other motifs. The regions it matches are close (14 and 37 nucleotides) to the regions matched by motif B. Therefore we hypothesize that this motif might be associated to additional regulatory elements.

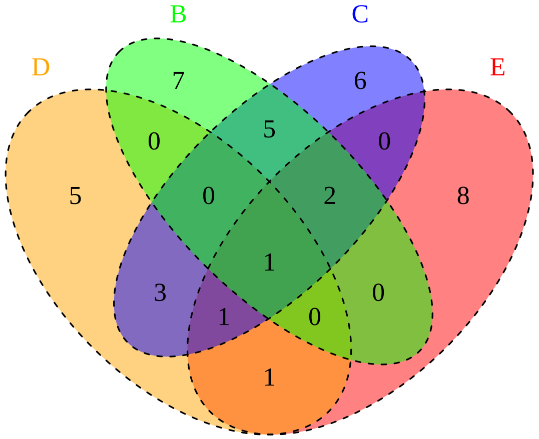


FIGURE 3.9: **Shared genes.** Presence of binding motifs A, B, C and D in gene upstream regions. See Figure 3.7 for Legends A, B, C and D motif description.

Scalability of network visualization

SyncVis scales quite well for visualizing synchronously large networks (i.e. networks with a few thousands nodes and edges). In other, words it is possible to upload multiple networks of these sizes to Cytoscape and then select a specific nodes. SyncVis can then successfully highlight these nodes on all networks.

In order to demonstrate this scalability, we have constructed a synchronous set of 11 networks on a ordinary desktop computer and then upload 11 gene identifiers from a file that were automatically in all networks. This visualization is presented in Figure 3.10. This construction is presented in detail in Additional file 7.

However it is good to keep in mind these networks tend to be clumsy, so it is not easy to browser them. If the user wants to gain detailed biological insight from them, then perhaps she should restrict to specific sub-networks such as specific signaling pathways presented in the “Synchronous visualization of differentially expressed genes under *S. aureus* infection on human and mouse signaling pathways” section.

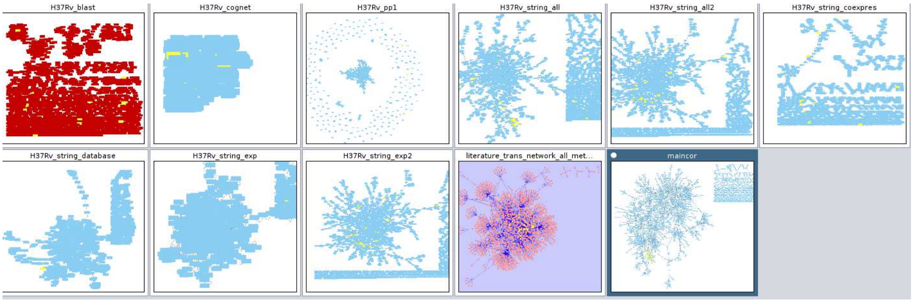


FIGURE 3.10: **Scalability of network visualization.** This Figure illustrates a synchronous visualization of 11 big networks. The selected nodes are highlighted by yellow in all networks. The exact sizes of the networks are displayed in Table 3.2.

Comparison with other tools, limitations and future directions

SyncVis is an integral part of SyNDI. SyncVis uses Cytoscape core for visualizing multiple networks. All generic development work done in the Cytoscape community will thus automatically be manifested in SyncVis. Moreover, the user of SyncVis can easily use other Cytoscape apps; for example there are some apps for advanced network visualization such as yFiles Layout Algorithms (`https://apps.cytoscape.org/apps/yfileslayoutalgorithms`), network comparisons [199, 287] and most importantly for network based biological analysis such as the ones illustrated in [478]. Tools like NAViGaTOR [82], Pajek [49] or igraph [120] are ideal for visualizing and/or analysing large networks but we have decided to implement SyncVis as a Cytoscape app due to the huge community effort behind Cytoscape and the continuous community support to biology oriented applications.

For the time being, SyncVis contains an automatic connection to only a few selected tools on the Galaxy platform. Some of the tools deployed in the presented use cases require collection of information that specifically relates to the studied organism, such as GO gene annotation and upstream sequence information for each gene. This information is derived from the genome but requires additional bioinformatics analysis or database mining for each organism and different tools have to be used for fungi, bacteria, mammals and so forth. We have chosen not to include the retrieval of this information as part of the SyNDI framework, which might limit its application. A potential future direction is to connect SyncVis to tools for genome analysis such as GenomeSpace [398] or SAPP [273]. The modular design of SyNDI allows addition of more of these tools. Finally, in the current version, the user has to install additional software components to use SyNDI's workflow. This could be streamlined by providing a script that installs all of these components.

TABLE 3.2: **The sizes of networks in Figure 3.10.** The first column contains the network names, the second columns contains the number of nodes in the network, and the third column contains the number of edges in the network.

Network name	#nodes	#edges
H37Rv_blast	1705	2855
H37Rv_cognet	894	51718
H37Rv_pp1	1681	7446s
H37Rv_string_all2	2919	7222
H37Rv_string_all	3020	7562
H37Rv_string_coexpres	1159	1904
H37Rv_string_database	853	4216
H37Rv_string_exp	451	1415
H37Rv_string_exp2	3020	7562
literature_trans_network_all_methods	1304	2132
maincor	2616	6072

Conclusions

Here we have presented SyNDI, which is a framework that connect a user-friendly Cytoscape application for synchronous network representation to advanced additional analysis tools for example through a Galaxy interface.

We have showed the potential of such a framework through three use cases. Firstly we have shown how the synchronous SyNDI framework facilitates differential network analysis and how dedicated layouts can help pinpoint altered metabolites' connectivity patterns at different levels of cardiovascular disease risk. Specifically such representations clearly emphasizes the altered interplay between amino acids and glucose at high latent risk.

Secondly, we have used SyNDI to compare common inflammatory response pathways in human and mouse by synchronous visualization of differentially expressed genes. We have visualized *S. aureus* infection transcriptomics data from human and mouse on signaling pathways. Most interestingly, inspection of the insulin signaling pathway a potential role of TLR2, which can induce SOCS3, in induction of inflammatory pathways in *S. aureus* infection even though there are so far very limited amount of studies to explain why insulin signaling is regulated in bacterial infection.

Finally, we have shown how SyNDI can be used to explore and better understand complex regulated systems such as ESX-1 and associated virulence proteins in *M. tuberculosis*. In addition we were able to detect multiple and related binding motifs within the DosR regulon which have not yet been described in the literature, including a motif that we hypothesize it is related to *M. tuberculosis* SigE.

Galaxy enables further development of SyNDI, so that additional analysis modules can be added and complemented with network visualization. Here only omics data has been used, but other data types (such as text mining results) and dedicated analysis tools can be seamlessly integrated within the framework. Users can also easily customize SyNDI for their needs as they can incorporate additional datasets to Galaxy and networks for visualization.

SyNDI provides a framework to visually inspecting local connections from multiple networks, regardless of their origin. Additionally, SyNDI integrates network visualization and and analysis through Galaxy. This represents major advantages with respect to the use of the separate tools in isolations. First of all there is an increase in usability, as the user can easily run analysis by selecting nodes on networks without complicated file handling (e.g. copy-pasting rows and columns from an Excel sheet to another). The second major advantage is that SyNDI and most important, the Galaxy interface, allows the development of analysis workflows so that in-silico analysis can be stored and re-used upon addition of new datasets.

Availability and requirements

The source code can be found at `https://gitlab.com/elindfors/syndi` and a link to the online user manual can be found at Cytoscape App Store `http://apps.cytoscape.org/apps/syncvis`.

Funding

This work was supported by the European Commission-funded FP7 project INFECT (contract number: 305340).

Chapter 4

Regulation of three virulence strategies of *Mycobacterium tuberculosis*: A success story

Adapted from:

Niels Zondervan, **Jesse CJ van Dam**, Peter J Schaap, Vitor AP Martins dos Santos, and Maria Suarez-Diez. "Regulation of Three Virulence Strategies of *Mycobacterium tuberculosis*: A Success Story". In: *International Journal of Molecular Sciences* 2018, 19(2), 347.

Abstract

Tuberculosis remains one of the deadliest diseases. Emergence of drug-resistant and multidrug-resistant *M. tuberculosis* strains makes treating tuberculosis increasingly challenging. In order to develop novel intervention strategies, detailed understanding of the molecular mechanisms behind the success of this pathogen is required. Here, we review recent literature to provide a systems level overview of the molecular and cellular components involved in divalent metal homeostasis and their role in regulating the three main virulence strategies of *M. tuberculosis*: immune modulation, dormancy and phagosomal rupture. We provide a visual and modular overview of these components and their regulation. Our analysis identified a single regulatory cascade for these three virulence strategies that respond to limited availability of divalent metals in the phagosome.

Introduction

M. tuberculosis (*Mtb*) is the most successful known intracellular pathogen infecting roughly one third of the world population and killing about 1.3 million people in 2017 alone [322]. Treating *Mtb* infection is increasingly difficult due to increasing number of drug-resistant, multidrug-resistant and extensively drug-resistant strains [322]. In order to come up with new drug targets and treatment strategies, there is an urgent need to understand the molecular mechanisms supporting the success of this versatile pathogen. Here, we will review the regulation of three important survival strategies of *Mtb*: immune modulation, dormancy and phagosomal rupture [115, 191, 330].

Firstly, *Mtb* is a master in immune modulation. Its ability to interfere with host cell signalling pathways allows it to carefully balance production of cytokines involved in activation of the pro-inflammatory and anti-inflammatory response [140, 210]. By balancing the pro- and anti-inflammatory immune response, *Mtb* delays phagosome maturation, harvests essential nutrients and stimulates the formation of granulomas. At early infection states, these granulomas are initially dominated by alveolar macrophages and shield the bacteria from more effective immune cells [449].

Secondly, when residing in the hypoxic granuloma, *Mtb* enters a metabolically near inactive and non-replicating dormant state in which it is immune to most types of drugs [200]. *Mtb* manipulates the macrophages to accumulate lipids, providing it with the nutrients required to sustain dormancy for multiple decades [262, 373, 422, 442, 449].

Thirdly, *Mtb* has a highly regulated pore formation system that it uses to rupture the phagosome and gain cytosolic access, resulting into necrosis of the host cell and dissemination of the bacilli [429, 452].

The fine-tuned regulation of these three virulence strategies is what makes *Mtb* such a successful pathogen. A large body of literature exists on these virulence strategies and on their molecular components. However, there have been few attempts to provide a systems wide overview of these three virulence strategies, their molecular components and their regulation. Divalent metals play an important role in the regulation of some key aspects of these strategies [173, 255, 307]. Here, we will present an overview of their involvement in this regulatory process. Detailed inspection of available knowledge pinpoints a single regulatory cascade as a main control hub for these three virulence strategies, representing their interconnectivity as subsequent stages encountered in pathogen host interaction. A modular overview of the molecular components involved in divalent metal homeostasis and their components involved in these three virulence strategies can be found in Supplementary Files 1 and 2. In the following, we will discuss these components and the environmental cues that control them and we will highlight the role of divalent metals in the phagosome.

Divalent Metals at the Interface of *M. tuberculosis* Host Interaction

Divalent metals such as iron, zinc and manganese are required for proliferation and survival of all living organisms. Divalent metals appear, in all living beings, nearly exclusively as constituents of proteins and act as cofactors in many essential enzymes and environmental sensors [238]. Iron is the most commonly used divalent metal cofactor [238]. Iron containing enzymes are involved, among other processes, in electron transfer, maintaining redox balance and detoxification [374]. Manganese has the strongest affinity for ATP and is the preferred cofactor in cAMP production [380, 405]. Zinc is used as cofactor by numerous enzymes and DNA binding proteins and additionally functions to scaffold additional proteins [312].

To prevent growth of bacteria, the host uses high affinity iron binding proteins such as lactoferrin, ferritin and transferrin to keep concentration of free iron in the blood low, in the so-called iron sparing response [255, 282]. These proteins also bind other divalent metals such as manganese, albeit with lower specificity than iron. Similarly, calprotectin functions as high affinity calcium binding protein but also binds manganese, zinc and iron in the blood [302]. During infection, macrophages withdraw approximately 30% of the total circulating iron from the blood stream making macrophages environments rich in divalent metals [363]. Some intracellular pathogens use this defence mechanism to their advantage by stimulating phagocytosis by macrophages to get access to divalent metals and other nutrients. During initial infection, *Mtb* predominantly encounters resident, replicative alveolar macrophages populating the lungs which are rich in divalent metals while having reduced bactericidal abilities compared to other macrophages [363, 422].

Upon ingestion by a macrophage, *Mtb* is engulfed in a special compartment called the phagosome, in a process known as phagocytosis. The phagosome then fuses with vesicles containing enzymes and other proteins that facilitate bacterial digestion. Phagocytosis is a rapid process and leads to phagosomal-endosomal fusion in approximately 3–4 min, acidification of the phagosome within 23–32 min and fusion with lysosome in 74–120 min, based on experiments with epithelial macrophages [64]. However, *Mtb* blocks phagosome maturation in an early phase leading to fusion with early endosomes and a pH of approximately 5.5 [172].

The macrophage continuously exports divalent metals out of the phagosome via Nramp1 and Nramp2 in a pH dependent manner. Many cell types express Nramp2 while only macrophages express Nramp1. Nramp1 is mechanistically similar to Nramp2 but has a much higher specificity for manganese (Mn) compared to Nramp2 [172, 173, 364]. Mn is required as cofactor for the bacteria to break down oxidative compounds produced in the phagosome such as H_2O_2 [242, 255, 380]. Thus, restricting Mn availability in the phagosome by recruitment of Nramp1 is an essential defence against intracellular pathogens. Nramp2 functions optimally around pH 6, a condition found in the early phagosome while Nramp1 has an optimal activity at a pH of 4.5

Nramp1 is attached to the membrane of maturing phagosomes and is associated with increased recruitment of endosomes and/or lysosomes containing vacuolar V-H^+-ATPase, resulting in acidification of the phagosome from pH 6.5 to 5.5 [172, 480]. Nramp2 is regulated separately from Nramp1 and co-localizes with transferrin receptors to early endosomes as well as with V-H^+-ATPase. V-H^+-ATPase provides the electro-genic force needed for Nramp1 and Nramp2 to operate [385, 516]. Metal availability in the phagosome is tightly regulated by the host through the combined action of Nramp1 and Nramp2. Therefore, blocking phagosome maturation is an effective strategy to create an environment in which *Mtb* can outcompete divalent metal export from the phagosome. *Mtb* uses special high affinity siderophores (mycobactin) to gain access to divalent metals from both extracellular transferrin and the intracellular iron pool [363].

Within *Mtb* iron, zinc and manganese homeostasis are regulated by IdeR, Zur (previously known as FurB) and MntR respectively [312, 374, 375]. Ligation of Fe^{2+} to IdeR and Zn^{2+} to Zur stabilizes the formation of dimers that have strong affinity to binding sites involved in suppressing the genes in their respective regulons [307, 374, 391]. MntR in *Bacillus subtilis* contains two manganese binding sites as well as a dimerization site similar to IdeR and Zur [138]. There is a significant overlap between IdeR, Zur and MntR regulated genes, see Figure 4.1. An overview of the regulation of molecular components by divalent metal regulators, IdeR, Zur and MntR can be found in Supplementary Files 1 and 2. Each of these three regulators suppresses the main operon of genes coding for the ESX-3 secretion system and associated PE, PPE and Esx proteins homologues of ESAT-6 and CFP-10 (EsxA and EsxB) [375]. We will further discuss the ESX-3 transport system in a section below. In the following sections, we will discuss main characteristics of genes regulated by Fe, Zn and Mn respectively.

Iron Homeostasis and Redox Sensing

Mtb produces high affinity hydrophilic and lipophilic siderophores termed carboxy-mycobactin and mycobactin, respectively. Mycobactin can bypass the phagosome membrane to scavenge iron from the extracellular iron storage protein transferrin [308, 328, 363, 526]. In addition, *Mtb* actively synthesizes deoxy-mycobactin during iron starvation [314].

Mtb combines the expression of a dedicated iron acquisition machinery with cellular components involved in immune modulation. By limiting acidification of the phagosome, *Mtb* maintains favourable conditions in which it can outperform active export of divalent metals by the macrophages transporter Nramp1. *Mtb's* success in acquiring iron is illustrated by a 20-fold increase of iron concentrations in the phagosome between 1 and 24 h of macrophage infection [419]. However, high iron concentrations renders *Mtb* much more vulnerable to the formation of oxygen and nitrogen radicals upon phagosome maturation, as iron functions as a catalyst in the formation of radicals via

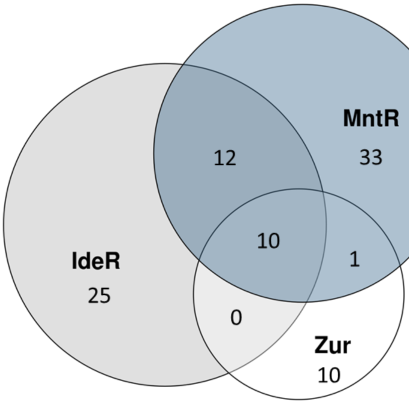


FIGURE 4.1: Number of genes in the IdeR, Zur and MntR regulons

the Fenton reaction [505]. Tight regulation of iron homeostasis is, therefore, essential, making IdeR an interesting drug target [417]. *Mtb* has adapted to deal with oxidative stress outside of the cell but is relatively vulnerable to endogenously generated oxidative stress in comparison to *Mycobacterium smegmatis* [505]. Due to this vulnerability, vitamin-C is an effective drug to combat *Mtb* in the early stage of infection by inducing the Fenton reaction in iron rich phagosomes [514]. The oxidative conditions encountered in the phagosome leads to oxidation of the intracellular iron pool. Oxidation of the iron pool derepresses IdeR regulated genes among which some are involved in virulence. Upregulating expression of virulence genes in low iron and oxidative conditions is a common response in intracellular pathogens and has been observed in *Shigella dysenteriae*, *Corynebacterium diphtheniae*, *Yersinia pestis* and *Yersinia pseudotuberculosis*, as well as in *Mtb* [303, 433].

The iron pool within *Mtb* and the phagosome functions as redox sensor to the oxidative conditions encountered in the early phagosome. In oxidative conditions, ferrous iron (Fe^{2+}) is oxidized to ferric iron (Fe^{3+}) [370]. Ferric iron does not bind to IdeR, leading to upregulation of IdeR suppressed genes in oxidative conditions [417]. Genes suppressed by IdeR code for proteins involved in siderophore synthesis (*mbtA-G*), secretion (*mmpL4/5*, *mmpS4/5*) and uptake (*irtAB*) as well as 11 genes coding for the ESX-3 secretion system, among others [163, 438, 447]. Even though IdeR mainly functions as iron dependent repressor, IdeR also induces transcription of four genes. Among the induced genes, *bfrB* and, to a lesser extent *bfrA*, code for mycobacterial ferritin-like iron storage proteins, which prevent overload of iron within *Mtb* [374, 416]. Analysis of the promoter region of *bfrB* revealed it contains two tandem IdeR binding sites involved in alleviating repression by Lsr2. Lsr2 is a histone like regulator that binds AT-rich regions virulence islands, including those coding for ESX-1, espACD and PDIM coding genes, acting as a global regulator to aid in the adaptation to extremes in oxygen availability [113, 114, 179, 365, 416]. Combined regulation of *bfrB* by Lsr2 and IdeR, suggests iron storage by BfrB is suppressed by Lsr2 during infection under changing oxygen conditions unless IdeR detects availability of intracellular ferrous iron which indicates a lack of oxidative conditions. Under low iron conditions, BfrA is required to mobilize stored iron. On the other hand, on high iron conditions, BfrB is needed for iron storage [268]. BrfB was shown to be required for the long term persistence of *Mtb* in iron-starved granulomas [282].

Iron homeostasis is an essential process for bacterial survival, therefore its cellular components are interesting drug targets. This was shown in a knockout study of the *mmpS4/5* siderophore secretion, which resulted in limited intracellular availability of iron as well as intracellular accumulation of siderophores toxic to *Mtb* [248]. Another interesting drug target is HupB, a nucleoid-associated protein that protects *Mtb* against reactive oxygen species, regulates siderophore synthesis and was proposed to facilitate transfer of iron from ferri-carboxymycobactin to mycobactin [309, 376]. HupB stimulates transcription of its own operon in the absence of IdeR-Fe^{2+} [309].

IdeR also regulates genes involved in response to oxidative and acidic stress, among which the two-component system PhoPR. Two-component systems contain a histidine kinase sensor that senses specific environmental stimulus and a response regulator that gets phosphorylated by the sensor upon specific environmental stimuli. Many two-component regulators, among which PhoPR, also regulate their own operon [201]. Presence of multiple binding sites allows both positive and negative regulation depending on the concentration and phosphorylation state of the response regulator, as is the case for PhoPR [202, 213]. PhoPR is the main regulator of the oxidative and acidic stress response but also it is the initial step in a regulatory cascade controlling pore formation and phagosomal rupture. Six putative IdeR binding sites upstream of the *phoP-phoR* operon were located, of which five were observed to bind IdeR in the presence of iron [317]. This points to a possible link between iron homeostasis and PhoPR regulation of the oxidative stress response and virulence genes.

Nevertheless, the exact role of IdeR in upstream binding of PhoPR remains to be determined.

Oxidation of the iron pool is also sensed by proteins containing iron-sulphur clusters such as the enzyme aconitase (Acn) and the regulators FurA and WhiB1-7. Acn catalyses the isomerization of citrate to isocitrate via cis-aconitate in normal conditions. However, in low iron or oxidative conditions it binds to and suppresses translation of IdeR-mRNA while increasing translation of TrxC-mRNA [39]. The function of Acn as redox sensitive translational regulator is conserved in many organisms [370, 384].

FurA (ferric uptake regulator A) regulates the oxidative stress response by modulating expression of the operon coding for FurA and the KatG catalase [418]. KatG is essential for the breakdown of H_2O_2 radicals formed upon phagosome endosome fusion and activates the anti-cell-wall drug isoniazid. Recently, transcriptional activation of *furA-katG* was found to be regulated by RbpA, which is induced by H_2O_2 in a SigE dependent manner [232].

A third iron sensitive regulator is WhiB7. WhiB proteins are iron-sulphur cluster-containing redox-sensing transcription factors. WhiB7 expression is auto-regulated by binding to its own promoter in response to antibiotics or redox stress [542]. An 80-fold upregulation of WhiB7 was observed upon treatment with antibiotics that bind to the 30S ribosomal subunit such as kanamycin and streptomycin [542]. WhiB7 is upregulated by iron starvation and was shown to induce transcription of *eis* and *tap* [408], two antibiotic resistance genes. Upregulation of *eis* increases secretion of IL-10 and slightly represses production of TNF-α by the host. IL-10 and TNF-α are involved in the anti-inflammatory and pro-inflammatory responses respectively [428].

In summary, oxidation of the iron pool is an important environmental cue to activate molecular components involved in iron sequestering, immune modulation and virulence. IdeR, FurA, Acn, WhiB7, Lsr2 and SigE are all involved in the response to the oxidative conditions encountered in the phagosome and subsequent adaption through expression of a vast repertoire of molecules involved in iron homeostasis as well as genes involved in modulation of the immune response.

Manganese Homeostasis and cAMP Production

Manganese is one of the most abundant metal elements in nature [164]. Mn is involved in enzymes of diverse functionality such as photosynthesis and detoxification: Mn is used as cofactor for both synthesis and degradation of H_2O_2, superoxide and radicals [255]. The oxidative burst is a very effective bactericidal process to defend against intracellular pathogens such as *Mtb* and *Y. Pestis* [97, 113, 279]. As previously stated MntR is a regulator of Mn homoeostasis, however MntR is dispensable for *Mtb* growth in human and/or mice macrophages due to the limited availability of Mn in the phagosome. Manganese transport on the other hand is required for virulence and to break down oxygen radicals [375]. *Mtb* contains two superoxide dismutases, SodA and SodC. SodA uses manganese as preferred cofactor and requires CtpC for metalation and export to the phagosome. Interestingly, *ctp*C transcription is induced in the presence of PhoP, while *sodA* is predicted to contain upstream cAMP-CRP binding sites implicating it in its regulation [8, 201]. CRP is a cAMP dependent regulatory protein.

Another role of Mn we would like to discuss here is the Mn dependent activation of cAMP production in the early phagosome which was first proposed by S. Reddy et al. in 2001 [405]. S. Reddy and co-workers studied the kinetics of membranes containing *Mtb* adenylyl cyclase CyA (Rv1625c). Their study revealed that the Michaelis-Menten constant (Km) for Mn-ATP is

70-fold lower than for Mg-ATP. This results in a 47-fold activation by 1 mM Mn-ATP compared to 1 mM of Mg-ATP at physiological conditions [405]. Mn is also essential for the CRP regulated, virulence associated type III phosphodiesterase Rv0805 [127, 325].

During infection, intracellular cAMP concentration increases 50 fold and this is associated with a decrease in pH from 6.7 to 5.5 [32]. Among the 15 Adenylate Cyclases (AC) present in *Mtb* H37Rv, CyA has the highest measured cAMP production while AC (Rv1264) functions optimally at pH 6, which is typically found in early phagosomes [32, 145]. *Mtb* was shown to secrete cAMP in a burst into the macrophage cytosol, resulting in a 10-fold increase in the host's TNF-α concentration, an important inducer of granuloma formation [5]. Rv0386 is needed for this cAMP burst [5].

The MntR regulon contains *mntH* (Rv0924c), coding for Mramp, an Nramp homolog that imports manganese (Mn) in a pH dependent manner; *mntABCD* (Rv1283c-Rv1280c) coding for an ATP dependent manganese transporter and *Rv2477c* coding for a manganese dependent ATPase which optimally functions at pH 5.2 [126]. Interestingly, Rv2477c was postulated to be involved in resistance to tetracyclines and macrolides [126]. Additionally, MntR and Zur regulate *Rv2059-Rv2060* coding for two components of an incomplete ABC transporter of unknown function. Therefore, it is more likely that this transporter is involved in transporting other divalent cations like Co^{2+}, Cu^{2+} or Ca^{2+} to substitute Mn and Zn in some conditions. A second possibility is that this operon codes for a divalent cation exporter to counter the side effect of unwanted uptake of divalent cations such as Cu^{2+} by the high expression of manganese and zinc transporters [375]. Manganese uptake plays an important role in virulence of many bacteria. For instance, supplementing *Salmonella typhimurium* with manganese prior to infecting macrophages, decreased its lethal dose 50-fold [413]. Similarly, manganese acquisition in the gut was shown to allow *S. typhimurium* and *Salmonella enterica* to evade neutrophil killing by calprotectin and reactive oxygen species, while patients with mutations in manganese transporter Nramp1 were shown to be much more susceptible to pathogens such as *Mtb* [6, 97, 113, 139, 172, 380].

MntR regulates WhiB6 which regulates *espACD* and some DevR (previously known as DosR) regulated genes [379]. DevR is the main regulator of dormancy and *espACD* is involved in pore formation [105] and will be discussed below. The WhiB6 iron sulphur cluster is necessary for the negative control of the DevR regulon and positive control of the ESX-1 secretion system, whereas apo-WhiB6 induces the DevR regulon and suppresses ESX-1 expression in *M. marinum* [105]. A model was proposed where holo-WhiB6 positively regulate ESX-1 operon while upon reaction with reactive oxygen species and NO, apo-WhiB6 and WhiB6-DNIC are formed respectively. Both apo-WhiB6 and WhiB6-DNIC activate DevR regulated genes to shift metabolism and maintain energy and redox homeostasis [105].

MntR interacts with the toxin-antitoxin system RelJ and RelK in which MntR functions as antitoxin [274, 530]. Additionally, VapBC26 and VapB30 toxin-antitoxin system both requires Mg or Mn for their ribonuclease activity, which inhibits growth [261, 293]. These results indicate Mn might function as environmental cue in the regulation of growth.

Zinc Homeostasis

The third and final divalent cation we would like to discuss is zinc, the only redox stable divalent metal of the three. As previously stated, zinc homeostasis is regulated by Zur (FurB), a Zn^{2+} dependent repressor. Zur knockout studies identified 32 genes that are upregulated in the *zur* knockout mutant of which 24 belong to eight transcriptional units that were shown to be directly regulated by Zur [312]. Zur expression levels are regulated by SmtB encoded by an upstream gene, which is co-operonic with *zur*. SmtB functions as a repressor which is deactivated upon binding to Zn^{2+} [312].

There are three possible zinc uptake systems regulated by Zur. Firstly, Zur regulates the *sitABC* like genes (*Rv2059-2060*), which are also regulated by MntR that were previously discussed. This suggest that this transporter might function as Zn importer [78, 380, 439]. Secondly, Zur regulates *Rv0106* coding for a protein similar to the *B. subtilis* putative zinc low-affinity transporter YciCas [439]. Thirdly, EsxG-EsxH proteins were shown to be able to bind zinc, which might implicate them in zinc transport [237].

Other interesting targets of Zur are five genes coding for ribosomal proteins that can function in the absence of zinc, in contrast to their zinc dependent counterparts which normally bind to the 30S ribosomal subunits [180, 312]. Although Zur was found to be able to positively regulate some genes in other pathogenic bacteria via repression of non-coding small RNAs, no such regulation was found in a *zur* knockout *Mtb* mutant [307].

ESX-3 Secretion System

The ESX-3 secretion system is the only one of the five ESX systems that is essential for in vitro growth of Mtb [85, 231]. ESX-3 is involved in divalent metal homeostasis and immune modulation. ESX-3 is involved in divalent metal homeostasis and immune modulation. ESX systems secret extracellular proteins [331, 451].

Regulatory binding site for all three divalent metal regulators IdeR, Zur and MntR can be found in the ESX-3 core operon promoter [237, 438], as summarized in Table 4.1. The triple control of ESX-3 might allow *Mtb* to switch partly to other divalent metals in the absence of one of these three. This hypothesis is supported by the observation that siderophore knockout mutants low in iron contain much higher zinc concentrations [516]. However, many ESX-3 associated genes are regulated by only one or two of these regulators, indicating dedicated roles in homeostasis of specific metals [503].

All three divalent metal regulators regulate EsxG and EsxH which play an essential role in secretion of PE and PPE proteins [503]. PE and PPE proteins comprise nearly 10% of the coding potential of the *Mtb* genome and, for many of them, immune modulating properties have been reported [499]. A

TABLE 4.1: Suppression of ESX-3 core genes and associated genes by IdeR, Zur and MntR.

Gene	IdeR	Zur	MntR
esx3-operon^1	-	-	-
esxG-esxH	-	-	-
esxQ		-	
esxR-esxS		-	-
esxW			-
ppe3		-	-
ppe4-pe5	-	-	-
ppe9	+		
pe13		^{2}	-
ppe19			-
ppe20			-
ppe37	-		
ppe38		^{2}	
ppe48		-	
pe_pgrs61			-

Plus symbols (+) indicate positive regulation, while minus symbols (−) indicate negative regulation. [1] *Rv0282-Rv291*; [2] Reported as Zur regulated by Maciag et al. based on direct experimental evidence on two conditions [312]; predicted not to be in the Zur regulon through a large scale analysis of transcriptomics datasets and analysis of binding sites in upstream sequences [123]

large number of studies exist on the immune modulating properties of ESX-3 secreted PE and PPE proteins [85, 122, 147, 299, 334, 426, 499, 503]. The ESX-3 secreted protein pair EsxG-EsxH, targets the endosomal sorting complex to impair fusion of the phagosome with the lysosomes, while increasing association with the endocytic pathway leading to fusion with transferrin containing vesicles [85, 237, 331]. PE5-PPE4 were found to be critical for the siderophore-mediated iron-acquisition functions of ESX-3 [503]. PPE38 inhibits macrophage MHC Class I expression, dampens CD8+ T-Cell responses and was shown to be required for virulence of *M. marinum* [147, 334]. PPE37 was found to reduce the production of pro-inflammatory factors TNF-α and IL-6 [122]. PE_PGRS61 binds TLR2 in a Ca^{2+} dependent manner, leading to increased IL-10 production. Finally, PE5 and PE15 trigger activation of the host MAP kinases required for IL-10 production [426, 499]. IL-10 is an important anti-inflammatory cytokine. IL-10 reduces the expression of *iNOS*, limiting production of nitric oxide (NO) in the phagosome [85, 499]. Enhanced IL-10 expression plays an important role in inhibiting early protective immunity and blocking phagosome activation [121, 316]. In addition, a direct role for IL-10 in *Mtb* reactivation has been observed [121]. Interestingly, IL-10 also modulates lipid metabolism by enhancing uptake and efflux of cholesterol in macrophages [121, 217, 316]. *Mtb* is known to induce foamy macrophages using immune modulating proteins as well as secreted lipids. This leads to deregulation of the macrophages lipid metabolism via the macrophages' lipid-sensing nuclear receptors PPARγ and TR4 [316, 422]. One study reported observing *Mtb* to exploit host vesicle trafficking and lipid storage by recruitment of iron bound mycobactin to lipid droplets which move to the phagosome and discharge their content [308]. Another study found that *Mtb* uses membrane vesicles containing immune modulating molecules as well as mycobactin to interact with the macrophage during infection [394]. Further research is needed to investigate the proposed synergy between modulation of host vesicle trafficking, lipid acquisition and iron acquisition.

Three Main Virulence Strategies of *Mtb*

The three virulence strategies discussed in this review, namely immune modulation, dormancy and phagosomal rupture, represent subsequent stages in *Mtb*-host interaction. These strategies extend and complement each other, which is reflected in their regulation. While many pathogens directly express components involved in phagosomal rupture, *Mtb* keeps a low profile and activates key virulence strategies, such as phagosomal rupture, only when immune modulation fails and the phagosome becomes inhospitable. However, immune modulation also complements phagosomal rupture and dormancy, since immune modulation leads to conditions, such as granuloma formation and cholesterol accumulation, which are needed to prepare *Mtb* for dormancy and phagosomal rupture.

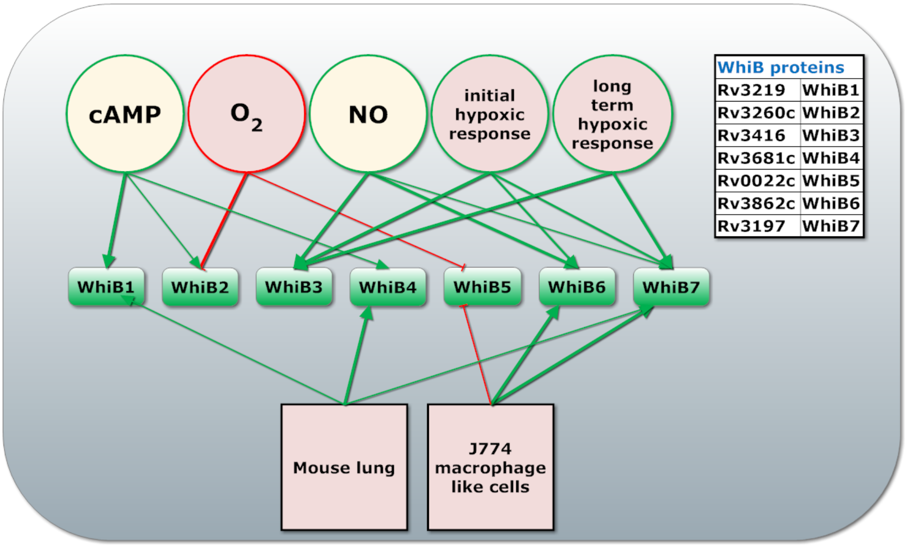
Immune Modulation

Mtb uses a number of virulence proteins, complex lipids and secreted metabolites, to modulate the immune response and arrest phagosome maturation to prevent fusion with late endosomes and lysosomes [32, 257, 330, 331, 441, 513, 526]. In case of successful immune modulation, phagosome maturation is halted resulting in a pH of approximately 5.5 [172, 480]. The macrophage controls intracellular trafficking, including phagosome maturation, through 42 distinct Rab GTPases. Rab5 is associated with phagosomes immediately after phagocytosis and normally diffuses quickly, allowing Rab7 to associate to the phagosome, which allows fusion of the phagosome with lysosomes. Studies with *M. bovis* have shown that *Mycobacteria* halts phagosome maturation, by blocking vesicle fusion between stages controlled by Rab5 and Rab7, with no Rab7 being accumulated in macrophages even after 7 days [513]. Similarly, for *Mtb* Rab7 was shown to be recruited by the phagosome but its premature release prevents fusion of the phagosome with late endosomes [211, 441].

In addition to the earlier discussed ESX-3 secreted proteins, several other proteins and molecules are involved in blocking phagosome maturation. Secreted tyrosine phosphatase (PtpA) is involved in the exclusion of the vacuolar V-ATPase, thereby preventing acidification and fusion with lysosomes [493, 526]. cAMP secreted by *Mtb* blocks phagosome lysosome fusion by inhibiting actin assembly [257]. Additionally, a number of virulence lipids interfere with the phagosome's Golgi trafficking, needed for maturation of the phagosome [211, 353]. Among these virulence lipids are monomycolate, dimycolate, sulpholipid-1, diacyl trehalose, polyacyl trehalose as well as phthiocerol dimycocerosate (PDIM). Of these lipids, PDIM was shown to play a role in phagosomal rupture and will be discussed in the section below.

Mtb is very successful in balancing the expression of molecular systems involved in activating the pro- and anti- inflammatory responses of the host to direct the immune response to favourable conditions for its survival. *Mtb* achieves this balance through multitude sensors and that integrate many environmental cues. One important family of regulators involved in sensing internal conditions are the iron-sulphur cluster containing WhiB family of regulators, already mentioned in the section on iron homeostasis. Different WhiB regulators have different redox potential and sensitivity to oxidative agents such as O_2 and NO and for some, thioredoxin like protein disulphide reductase activity has been reported [10, 288, 460, 542]. Many *whiB* genes are regulated by cAMP-CRP [542], as summarized in Figure 4.2.

WhiB1 is an essential regulator that senses NO, is regulated by cAMP-CRP and is associated with resuscitation [403, 460]. WhiB4 is associated to the oxidative stress response while WhiB5 is required for resuscitation [94, 102]. DNA binding has only been experimentally proven for WhiB1, WhiB2, WhiB3, WhiB6 and WhiB7 [105, 542]. Interestingly, WhiB1-3 are induced during infection and, upon nutrient limitation, by exogenous cAMP. This indicates they are involved in sensing the redox state of *Mtb* [453]. For WhiB1-3 it was shown that their DNA binding ability is enabled by NO by bringing their iron-sulphur cluster in their nitrosylated or apo-form [473, 542]. *whiB2* and *whiB3* are down regulated in presence of O_2 while *whiB3*, *whiB6* and *whiB7* are upregulated in the early or late hypoxic response. Of the *whiB* genes, *whiB7* is most upregulated in the macrophage with a 13 fold induction while being 80 fold induced by antibiotics that bind the 30S ribosomal unit [288]. WhiB3 senses NO and O_2 via its iron-sulphur cluster [279] and regulates genes involved in assimilation of propionate, a by-product of cholesterol degradation, into virulence lipids [2, 412, 454, 543]. Virulence lipids regulated by WhiB3 include sulfolipids, diacyltrehaloses and polyacyltrehaloses, which results in both higher pro- and anti-inflammatory cytokine levels and function as redox sync [181, 454]. WhiB3, PhoP and Lsr2 bind to and regulate the *whiB3* operon. MprAB might induce *whiB3* through upregulation of Rv0081, which was predicted to induce the *whiB3* operon [181]. In addition, WhiB3 together with DevSTR regulates expression of *tgs1* which is needed for the production of triacylglycerol, a storage lipid without which *Mtb* cannot resuscitate from dormancy [262, 279, 427]. WhiB1 is associated with resuscitation as it



WhiB proteins	
Rv3219	WhiB1
Rv3260c	WhiB2
Rv3416	WhiB3
Rv3681c	WhiB4
Rv0022c	WhiB5
Rv3862c	WhiB6
Rv3197	WhiB7

FIGURE 4.2: **WhiB1-7 transcriptional response to environmental stresses.** Proteins from the WhiB family are presented in the squares. The circles in the top indicate environmental cues (O_2, NO, cAMP availability) or infection stages (initial or long term hypoxic response). Squares represent different environments (mouse lung and JJ774 macrophage like cells). Arrows indicated regulation (green for induction, red for inhibition of transcription) with the line width indicating the strength of the interaction based on the fold change of their transcript level in a given conditions [288, 542].

induces transcription of *whib1*, *rpfA*, *ahpC* and *groEL2* in the absence of NO upon upregulation of WhiB1 by cAMP-CRP [460]. Interestingly, WhiB1 also interacts with GlgB, which is essential for optimal growth of *Mtb*, by reducing intramolecular disulphide bonds [94, 460, 542].

For a full review of WhiB proteins we refer to the excellent paper by Larsson et al. [288]. For a review of the function of WhiB like proteins and a network view of WhiB1-3 regulated genes and their connection to other virulence factors such as cAMP and CRP we refer to the review by Fei Zheng et al. [542]. An overview of WhiB regulators and the environmental cues they respond to can be found in Figure 4.2.

Two highly regulated virulence systems are EspACD, involved in phagosomal rupture and GroEL2, an abundant chaperonin involved in blocking apoptosis. Regulation of GroEL2 is summarized in Figure 4.3. GroEL2 is a highly antigenic gene and is associated with increased release of IL-10 and TNF-α which is also associated with cAMP secretion into the cytoplasm of the macrophage [5, 32, 188, 257, 473]. GroEL2 forms a dimer and is normally associated to the cell wall. However, Hip1 cleaves cell wall associated GroEL2 to form monomers that are able to cross the phagosome membrane and inhibit apoptosis by interacting with mitochondrial mortalin [251, 349]. In this way, Hip1 modulates the macrophage responses by limiting macrophage activation and dampening the activation of TLR2-dependent pro-inflammatory responses [349]. Interestingly, Hip1 has also been reported to function as lipase, making the proteolytic function of Hip1 somewhat disputed [174]. *Mtb* inhibits apoptosis of the macrophage through aggregation of mitochondria around the phagosome and increased activation of mitochondria resulting in limited cytochrome C release, an important inducer of apoptosis [243].

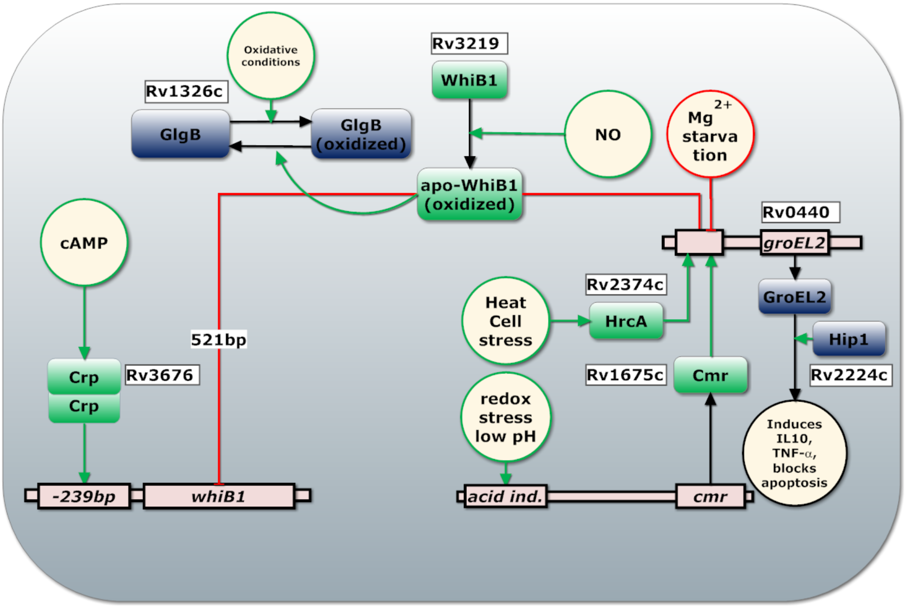


FIGURE 4.3: **Regulation of GroEL2.** Squares represent proteins, circles represent pools of simple chemicals, environmental cues or factors. Green lines indicate induction of transcription while red lines indicate inhibition of transcription. Black lines indicate causal effects.

CMR and HrcA positively regulate *groEL2* expression upon acidic and anaerobic stress [473, 476]. CRP induces *whiB1* expression in presence of cAMP while WhiB1 represses its own operon as well as *GroEL2* in the presence of NO [4, 473]. GroEL2 is therefore only expressed in the presence of cAMP or pH and redox responsive transcription factor CMR or heat stress, while NO is absent (Figure 4.3). GroEL2 expression is induced 24 h post infection but not at 2 h after infection while other CMR regulated genes, like *Rv1265* and *PE_PGRS6*, are induced at 2 h post-infection [189].

Phagosomal Rupture and Pore Formation

The second main virulence strategy deployed by *Mtb* is phagosomal rupture. A model of regulation of pore formation can be found in Figure 4.4.

ESX-1 and ESX-1 secreted proteins EsxA (ESAT-6) and EsxB (CFP-10) have been implicated in phagosomal rupture of many Mycobateria such as *M. marinum*, *M. kansii* and *Mtb* [30, 174, 301, 459]. The virulence lipid phthiocerol dimycocerosates (PDIM) and EsxA from *Mtb* were shown to interact with the host cell membrane and in concert, induce phagosome membrane damage and rupture in infected macrophages [30, 177]. A recent study reported that many claims about pore formation at neutral pH are due to contamination with detergent from the washing step [115]. The same study found membrane-lysing capabilities for EsxA only to occur below pH 5, to be contact dependent and accompanied by gross membrane disruptions rather than discrete pores. For the sake of simplicity, we refer here to the process of cytosolic access as *phagosomal rupture* although more research is needed to find out if cytosolic access is only achieved through lesions or also through formation of pores. Additionally there are reports of *Mtb* and other *Mycobacteria* to escape the phagosome [244]. However, the data generate by electron microscopy—the only direct approach—remains controversial.

The ESX-1 secretion system is involved in secretion of virulence proteins among which those shown to be involved in pore formation and phagosomal rupture EsxA (ESAT-6) and EsxB (CFP-10), secretion associated proteins EspA-D, EspF and secreted immune modulating PE and PPE proteins [111, 136, 142, 451]. Although EsxB is the main pore forming protein, other ESX-1 secreted genes are required for EsxB secretion and proper functioning of the ESX-1 secretion machinery. EspD stabilizes the extracellular levels of EspA and EspC and it is required for EsxA secretion but does not require ESX-1 for its own secretion [104]. Secretion of EspA, EspC, EsxA is codependent on each other, suggesting they might be secreted as a multimeric complex or that they are part of the secretion machinery itself [175, 241]. This theory is supported by a study showing that EspA forms dimers by disulphide bond formation after secretion; disruption of this disulphide bond affects cell wall stability as well as the functioning of the whole ESX-1 secretion system [184]. Recently, an EspC-multimeric complex was observed to form filamentous structure that could represent a secretion needle [29]. Inactivation of MyCP1 protease causes hyper-activation of ESX-1 while protease inhibition leads to attenuated virulence during chronic infection [306, 529]. A balanced activation and deactivation of ESX-1 through MycP1 proteolysis of EspB is required during chronic infection. MyCP1 and MyCP5 are required for stability of the ESX-1 and ESX-5 secretion complex respectively [525]. Without ESX-1, *Mtb* is unable to disrupt the phagosome membrane and make contact with the cytosol, leading to highly diminished pathogenicity [111].

ESX-1 and secreted factors EsxA and EsxB are regulated by the two-component systems PhoPR, previously mentioned. The importance of PhoP for virulence was confirmed in knockout studies that showed *phoP* knockout mutants

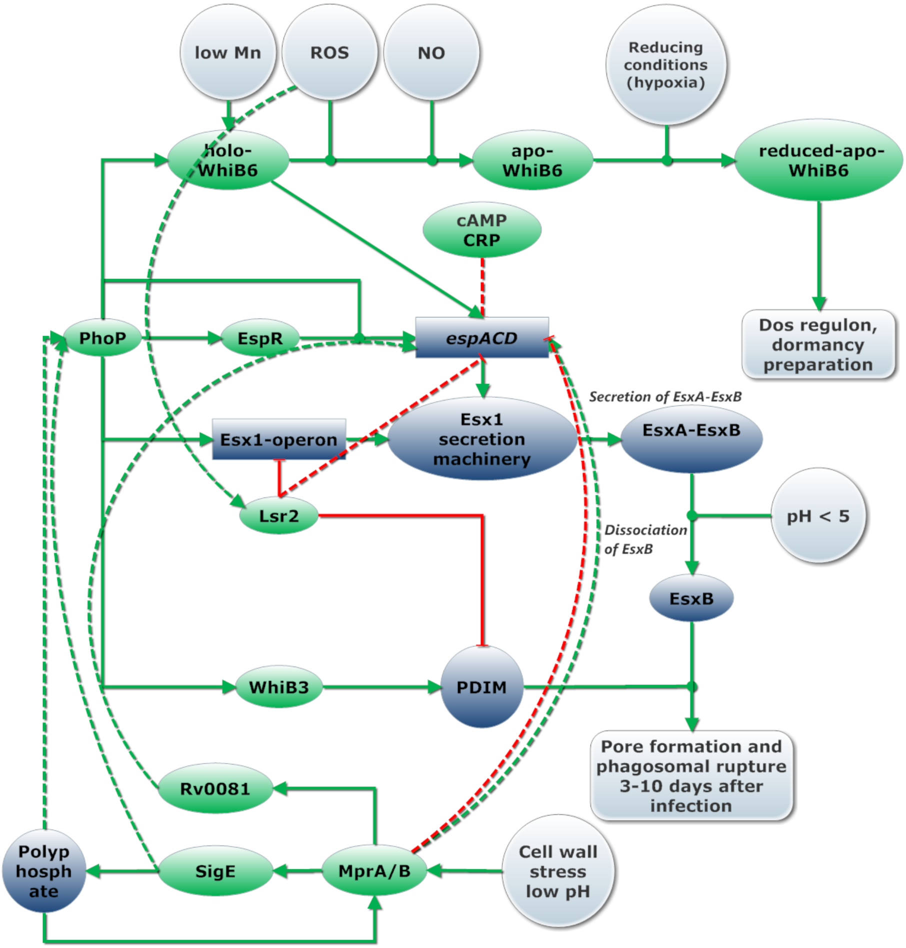


FIGURE 4.4: **Regulation of pore formation.** The circles represent environmental conditions. Arrows indicated regulation (green for induction, red for inhibition of transcription) with dashed lines for uncertain effects. Regulators are depicted in green, proteins and other molecules dark blue while operons are depicted in squares.

to be attenuated in mouse bone marrow derived macrophages, lungs, livers and spleen [456]. A single point mutation in *phoP* in *Mtb* H37Ra decreases the DNA affinity of PhoP and strongly contributes to the reduced virulence of this strain [387]. PhoPR regulated genes are upregulated in acidic and oxidative conditions encountered during the first two days of infection [419]. Recent studies show that PhoP interacts with SigE, which is upregulated in acidic pH and upon cell stress during the first three days of infection [41, 419]. Additionally, polyphosphate was needed for normal transcription of *phoP* as well as for transcriptional regulation of *sigE* by *MprAB*, although these results could not be reproduced [455, 481]. PhoP/R influences transcription of some 80 (according to some sources up to 150 [76]) genes directly as well as the transcription of a large number of genes indirectly via upregulation of WhiB6, EspR, DevS/R and WhiB3 [181, 201].

EspR is a transcriptional regulator upregulated by PhoP. EspR induces transcription of the *espACD* (*Rv3612-16c*) operon which is essential for phagosomal rupture and potential escape from the phago(-lyso)some [104, 184, 280]. PhoP, therefore, controls, directly (*espB/E-L*) or indirectly (*espA/C/D*), the 13 Esp proteins secreted by ESX-1 [61, 206, 280]. Recently it was found that holo-WhiB6 increases transcription of its own operon, the ESX-1 regulon and suppressed the DevR regulon, while apo-WhiB6 formed in anaerobic conditions and by prolonged exposure to NO, suppresses the ESX-1 regulon and induces the DevR dormancy regulon [105]. Interestingly, gene expression of EsxB by WhiB6 was highly induced after 30-min of NO exposure, decreased at 60 min and is highly reduced after 3 h of exposure to NO, indicating a short but intense activation of *espACD* by holo-WhiB6. Additionally binding sites for WhiB6 and Rv0081, a transcriptional factor regulated by MprAB, were predicted upstream of *espACD* [379]. These results suggest WhiB6, which is induced by PhoPR and MntR, plays an essential role in the regulation of phagosomal rupture and dormancy.

Induction of transcription of *espACD* by EspR requires the presence of PhoP [280]. In addition, MprAB, Lsr2 and CRP bind to the promotor region of *espACD* operon. Lsr2 represses transcription of both the *espACD* and the ESX-1 operon [379], while CRP binding inhibits expression of *espACD* [256]. Lsr2 binds to AT rich regions in the DNA, mostly virulence genes and is required for adaptation to extreme oxygen conditions [113, 179]. We hypothesize it is likely that Lsr2 represses the operon containing ESX-1 genes and *espACD* in oxidative conditions. This could serve to avoid further aggravation of the immune response. MprAB functions as a repressor of the *espACD* operon in cellular stress conditions, however MprA/B is also required for full expression of *espACD*. It is plausible to assume both positive and negative regulation by MprAB occurs based on the presence of multiple binding sites for MprA and two transcriptional start in the *espACD* operon [379].

Like the post-translational activation of GroEL2 by HiP1, membrane lysing capability of EsxA is activated only upon dissociation of EsxA from EsxB in acidic environment (pH 4–5) encountered when the phagosome matures. Acetylation of proteins in *Mtb* is cAMP dependent [174]. Acetylation improves dissociation of EsxA from EsxB at higher pH, a model where acetylation leads to reduced virulence was proposed [250]. Taken together, these studies indicate pore formation is strictly regulated, most likely only occurs when cAMP is depleted (no cAMP-CRP), might be inhibited by sudden changes in oxidative conditions (Lsr2), the phagosome acidifies and become hypoxic

(PhoPR) and pore formation is transiently induced by WhiB6 upon NO sensing [105]. MprAB further modifies activation of *espACD*, most likely both positively upon initial cell damage and negatively after prolonged cell stress and accumulation of polyphosphate, as indicated in Figure 4.2.

It should be mentioned that in addition to their role as regulators, Lsr2, CRP and EspR have also been characterized as nucleoid-associated proteins and as such might serve additional functions such as structuring the organization of the chromosome and, as has been shown for the ESX-1 and espACD operon, protecting DNA region from oxygen radicals [66, 179, 256].

Dormancy and Modulation of Granuloma Formation

The third virulence strategy deployed by *Mtb* is onset of dormancy. Dormancy is a non-replicating and metabolically near inactive state at which *Mtb* is immune to most drugs and can survive for decades [191, 262]. Dormancy occurs upon formation of mostly hypoxic granulomas [500]. Immune modulation that stimulates granuloma formation will therefore be discussed as a part of the dormancy virulence strategy.

When *Mtb* runs out of cAMP to secrete thereby suppressing phagosome lysosome fusion, the macrophages phagosome will fuse with late endosomes and lysosomes. As a result, the phagosome becomes increasingly hostile with lower pH, production of oxygen radicals and NO and fusion with vesicles containing lysozymes. In contrast, conditions encountered in granulomas are slightly more favourable for *Mtb.* Granulomas have reduced capacity to form oxidative radicals [442].

Mtb stimulates TNF-α production which leads to granuloma formation among others through secretion of cAMP into the cytosol [121, 402, 428]. A number of studies indicate that granuloma may be dispensable for preventing bacterial dissemination and may actually contribute to *Mtb* persistence and shield *Mtb* from more successful immune cells [373, 442, 449]. According to some models, *Mtb* containing granuloma's contain two types of macrophages: classically activated and alternatively activated [449]. *Mtb* shifts the macrophage population within the granuloma from being classically activated to alternatively activated macrophage which produce more anti-inflammatory cytokines (TGF-β, IL-10) and arginase. These diminish the amount of arginine available to iNOS, which results in reduced NO production [321, 442, 449]. A balance of pro-inflammatory and anti-inflammatory response via stimulation of TNF-α and IFN-γ production is needed for granuloma formation while IL-10 is the main negative regulator for this response, inhibiting formation of dense and hypoxic mature fibrotic granuloma's [121, 449]. Moreover, parameter sensitivity analysis for a granuloma model, showed IL-10 had the strongest influence on myofibroblast numbers at 300 days post infection and indicated IL-10 to play a major role in preventing differentiation of immune cells needed to develop protective immunity [121, 449].

A number of regulators allow *Mtb* to sense and adapt to hypoxia and maturation of the phagosome. The most important of these regulators is the two-component regulator DevRST which regulate genes coding for proteins that help *Mtb* prepare for dormancy and subsequent resuscitation [100, 187, 295]. A visual representation of DevRST response to environmental cues is present as part of Supplementary File 1. Both DevS and DevT can activate the DevR regulon through phosphorylation of DevR, which autoregulates its own operon through cooperative binding to two binding sites [100, 101, 187, 445]. DevT provides initial activation of the DevR regulon through phosphorylation of DevR and has the strongest sensitivity to CO and a weaker binding to NO and O_2 compared to DevS. DevS is sufficient for DevR activation after 5 days of infection [228, 277]. DevS phosphorylates DevR even in the presence of small concentrations of NO, negatively regulates the DevR regulon through phosphatase activity in the presence of O_2 while positively regulating the DevR regulon in reducing conditions [229, 265, 277].

Interestingly, even under non-inducing conditions and as such no phosphorylation of DevR, the DevR regulon is activated upon high enough concentrations of DevR, providing a possible explanation for enduring induction of the DevR regulon which might occur after prolonged autoactivation of its own regulon [445]. Among DevR regulated genes there are a few types of regulation. While some genes are strongly upregulated within a few hours of infection others are only mildly induced after 12–24 h in hypoxic and high NO conditions [101]. DevR and other two-component regulators can fine tune expression of genes through the presence of multiple binding sites and through phosphorylation which stimulates cooperative binding [100].

CO is released by the enzymatic activity of heme oxygenase-1 (HO-1) in lungs infected by *Mtb* [278, 448]. CO is an important dormancy inducer. Interestingly, *Mtb* has a unique heme scavenging and degrading systems that does not produce CO allowing *Mtb* to degrade heme without inducing the immune response or its own dormancy regulon.

Interestingly, there is evidence for two DevR regulated proteins to be involved in stabilizing the 30S ribosomal units under hypoxic conditions, while slowing down translation and protein synthesis in the process [86, 500]. *Mtb* uses lipids such as cholesterol as primary nutrient in this phase of infection via genes regulated by KstR and IdeR [181, 412], while increasing production of triacylglyceride (TAG) via *tgs1* which is under control of DevR and Whib3 [279].

Protein-protein interaction was observed between DevT and NarL, a lone two-component response regulator involved in nitrate and nitrite respiration in *Escherechia coli* [253, 292, 322]. Although the genes regulated by NarL in *Mtb* are unknown, we argue it is plausible that NarL is involved in regulation of *nirB*, *narU*, *narX*, *narU*, *nuoB* that are currently thought to be part of the DevR regulon.

NO is produced in the maturing phagosome and is an important dormancy cue sensed by DevT and DevS. *Mtb* expresses two truncated heme proteins, GlbN and GlbO, that help it detoxify from nitrate containing oxygen radicals such as NO while residing in the macrophage [26, 252, 350, 504].

Interestingly, *GlbN* is co-transcribed with *lpRl* coding for Lipoprotein LprI, which Acts as a lysozyme inhibitor [440]. The *GlbN-lpR1* Activated isoniazid inhibits truncated haemoglobin N that protects against reactive nitrogen and oxygen species as well as AcpM, which is required for mycolic-acid production [27, 215, 307, 390]. NO was found to help *Mtb* to survive in hypoxic and acidic conditions through anaerobic respiration [253, 486]. In addition, nitrate respiration plays an important role in dormancy and protection against hypoxic and acidic stress [267, 486].

Although DevRST and WhiB3 are involved in the preparation for dormancy, the enduring hypoxic response measured in a *devR* knockout mutant showed 230 genes to be differentially expressed with roughly half of them upregulated in in the first day of hypoxia and the other half only upregulated at 4 and 7 days of hypoxia [424]. These results indicate many genes involved in the enduring hypoxia response are not regulated by DevR. Resuscitation from dormancy is more elusive and less studied than dormancy. Resuscitation involves ClgR and both SigH and SigE are upregulated upon reaeration [510]. Also cAMP-CRP plays a role in resuscitation as it upregulates *rpfA* one of the five resuscitation promoting factors [4, 212, 540].

Success through Regulation of Virulence Strategies

Mtb anticipates changes in the interaction with the host by upregulating both internal and external sensors and regulators involved in sensing progression of the immune response. This allows the bacteria to adjust more quickly to progression of the immune response. External sensors involved in survival in the macrophage consists mostly of two-component regulators [76] (such as DevRST, PhoPR, MprAB, SenX3-RegX3, NarL) while for internal sensors, WhiB family proteins and regulators such as CRP and CMR are used. These sensors and regulators appear interconnected, thus forming a single regulatory cascade that controls the three virulence strategies, as represented in Figure 4.5. This regulatory cascade integrates many internal (cAMP, Mn, Mg, oxidative conditions and presence of NO) and external environmental cues (phagosome pH or cell wall damage) for fine-tuned regulation of key virulence systems. Examples of such virulence systems downstream this cascade are GroEL2, ESX-1, EsxAB and EspACD. Pore formation by EsxA depends on the regulation of ESX-1 by PhoP, Lsr2 and WhiB6 and on regulation of EspACD by Lsr2, EspR, PhoPR, MprAB, WhiB6 and Rv0081. Post translationally, pore formation by EsxA is regulated by proteolytic activity of MycP1, acetylation of EsxA and dissociation of EsxA-EsxB upon acidification of the phagosome [66, 105, 113, 174, 179, 250, 256, 379, 452, 459]. Similarly, GroEL2 is regulated by CRP, WhiB1, HrCA and Mg^{2+} starvation and post-translationally regulated by proteolytic cleavage by Hip1 [4, 189, 251, 349, 473, 476].

There is a great amount of overlap in this cascade, so that multiple environmental signals are considered in the regulation of these genes, as indicated in Figure 4.5. For example, some PhoPR regulated genes are predicted to have

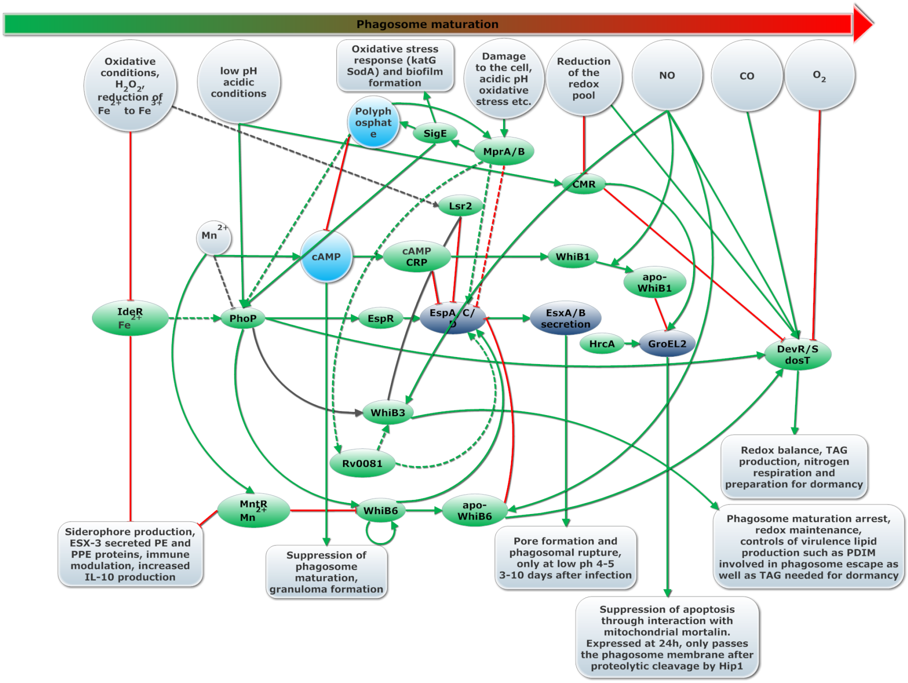

Phagosome maturation
Oxidative conditions, H_2O_2, reduction of Fe^{2+} to Fe^{3+}
low pH acidic conditions
Oxidative stress response (katG SodA) and biofilm formation
Damage to the cell, acidic pH oxidative stress etc.
Reduction of the redox pool
NO
CO
O_2
Polyphosphate
SigE
MprA/B
CMR
Lsr2
Mn^{2+}
cAMP
cAMP CRP
WhiB1
apo-WhiB1
IdeR Fe^{2+}
PhoP
EspR
EspA/C/D
EsxA/B secretion
HrcA
GroEL2
DevR/S dosT
WhiB3
Rv0081
Redox balance, TAG production, nitrogen respiration and preparation for dormancy
MntR Mn^{2+}
WhiB6
apo-WhiB6
Siderophore production, ESX-3 secreted PE and PPE proteins, immune modulation, increased IL-10 production
Suppression of phagosome maturation, granuloma formation
Pore formation and phagosomal rupture, only at low ph 4-5 3-10 days after infection
Phagosome maturation arrest, redox maintenance, controls of virulence lipid production such as PDIM involved in phagosome escape as well as TAG needed for dormancy
Suppression of apoptosis through interaction with mitochondrial mortalin. Expressed at 24h, only passes the phagosome membrane after proteolytic cleavage by Hip1

FIGURE 4.5: **Overview of the regulatory cascade that integrates environmental cues to active the immune modulation, dormancy and phagosomal rupture virulence strategies.** Arrows indicated regulation (green for induction, red for inhibition of transcription) with dashed lines for uncertain effects. Regulators are depicted in green, proteins and other molecules dark blue while operons are depicted in squares. The large arrow on the top represents the progression of the immune response.

cAMP-CRP binding sites [33]. These genes are upregulated upon oxidative stress and low pH but suppressed in the presence of cAMP-CRP, as is the case for *espACD* [411]. Some PhoPR regulated genes are also regulated by DevRST, WhiB3 and by MprAB. An even larger overlap exists in genes regulated by DevRST and MprAB, indicating integration of CO, NO, hypoxia and cell stress in the regulation of these genes [77, 377, 378]. We argue that based on the overlapping regulation of the three virulence strategies, these strategies extend and overlap each other. The order of activation of these strategies is likely to vary depending on the dynamics between *Mtb* and the host. Timing of specific virulence strategies also vary for different *Mtb* strains [244]. Some strains gain cytosolic access within hours of phagocytosis while others require 3–10 days [244, 452].

Pore or lesion formation is linked to immune modulation. Cytosolic access is need for secretion of cAMP and other immune modulating factors, such as

GroEL2, into the macrophage cytosol [244]. There are still many unanswered questions regarding the exact role and regulation of GroEL2. Firstly, it is unknown at which conditions proteolysis of GroEL2 by Hip1 (*Rv2224c*) occurs. Secondly, Hip1 was reported to mainly function as lipase in one study [174], further research is needed to confirm whether GroEL2 is a direct substrate of Hip1. Strict regulation of GroEL2 suggests it to have an important role in virulence.

Interestingly, there are many parallels in regulation of virulence systems between *Mtb* and other pathogens. Understanding *Mtb* as one of the most successful intracellular pathogens can therefore provide insight in common strategies deployed by intracellular pathogens. For instance, positive regulation of virulence genes by PhoPR and suppression by cAMP-CRP appears to occur in more pathogens. In *Y. pestis*, PhoP directly binds to and transcriptionally activates *crp* and *cyA* leading to merging of the PhoPQ and CRP-cAMP regulon [541]. Similarly, a major virulence island is positively regulated by PhoP while being suppressed by cAMP-CRP in *S. typhimurium* [247]. In *Mtb*, PhoPR regulates pro-inflammatory virulence genes such as the ESX-1 operon as well as genes involved in protecting against oxidative stress, when cAMP is depleted. cAMP does not only suppress phagosome maturation but also acts as an internal sensor of phagosome maturation, through pH dependent secretion of cAMP.

Some aspects in the regulation of PhoPR and cAMP in *Mtb* require more research. Firstly, the function of multiple IdeR binding sites upstream of the *phoPR* suggests complex regulation of the *phoPR* operon by IdeR and thus by iron bioavailability. Secondly, the exact cue for activation of PhoP remains unknown. Upregulation of *phoPR* in acidic conditions has been observed as well as under Mg^{2+} starvation, however this later observation could not be reproduced [2]. Transcriptional analysis of *Mtb* showed many genes in the PhoPR regulon to be upregulated during the first hours of infection (20 min to 2 h) while the phagosome acidified from pH of 6.5 to pH 5.5 [418]. PhoPR stimulates expression of aprABC, an Mtb specific pH sensing locus involved in the regulation of among others a number of PhoP regulated genes [2]. These results indicated PhoPR directly or indirectly senses pH. Recently, it was discovered that PhoP interacts with acid inducible extracytoplasmic sigma factor SigE, providing a possible explanation for activation of the PhoP regulon at low pH [41]. Extracytoplasmic sigma factors provide a means of regulating gene expression in response to various extracellular changes, hence their name.

Secondly, we argue entrance of *Mtb* in the early phagosome is likely to lead to higher abundance of Mn. Pathogenic *Mycobacteria* species such as *Mtb* and *M. avium*, have high manganese concentrations at 1 and at 24 h after infection compared to non- pathogenic *M. smegmatis* [516]. Mn availability might also be affected by Mramp, a pH dependent Mn H^{+} symporter with maximal activity between pH 5.5 and 6.5 matching the conditions found in the early phagosome. Mn is an important cofactor for cAMP synthesis and it is likely to increase cAMP production in the early phagosome. cAMP-CRP and PhoPR co-regulate virulence genes directly or via regulators such as WhiB6, which is linked to Mn deficiency. Based on the strong affinity of PhoP for Mn we hypothesize Mn might play a role in both cAMP and PhoPR regulation [6, 380]. Depletion of Mn and secretion of cAMP might lead to de-repression of cAMP-CRP suppressed genes such as *espACD* as well as activation of these genes through PhoPR.

Thirdly, polyphosphate is needed for optimal PhoP activation [455]. Polyphosphates are potent inhibitors of type III adenylyl cyclases in *M. bovis* which agrees with the opposing roles of cAMP-CRP and PhoPR in respectively inducing genes involved in the anti- and pro-inflammatory response in *Mtb* and other pathogens. Polyphosphate is implicated in the activation of PhoP and is part of one of two positive feedback loops in the regulation of *mprAB* and *sigE* [41, 455, 481]. Polyphosphates kinase production is conserved in all bacteria and is associated to induction of dormancy and activation of virulence genes in many pathogens [83]. Knockout polyphosphate kinases *ppk1* mutants, have reduced biofilm formation, are more susceptible to drugs and are impaired in growth in guinea pigs [110, 455]. Interestingly, SigE is involved in regulation of polyphosphate. MprAB and SigX3-RegX3, induce transcription of *sigE* upon cell wall stress or phosphate starvation, while anti sigma factor RseA binds to and neutralizes SigE in reducing conditions [318, 430]. RseA is degraded by ClpC1P2-dependent proteolytic activity depending on its phosphorylation by the eukaryotic-like Ser/Thr protein kinase PknB [318]. SigE, polyphosphate and MprAB are involved in a double positive feedback loops through polyphosphate and ClpC1P2 of which a visual model is provided by Manganelli et al. [318]. Polyphosphate functions as phosphate donor for MprAB under low ATP condition. Additionally, SigE regulates the transcription of the *furA-katG* operon in response to oxidative stress in *Mycobacteria* [232]. SigE knockout strains are strongly attenuated and a recent study shows a *sigE* knockout strain provide an even more effective live vaccine than BCG [502]. Taken together, these studies indicate SigE plays an important role in adapting to low pH, cell wall and oxidative stress through upregulation *furA-katG*, activation of some PhoPR induced genes, MprAB and inhibition of cAMP-CRP through polyphosphate production. The interplay of SigE, polyphosphate and the hypothesized role of Mn in PhoPR and cAMP regulation should be further investigated.

Another aspect we want to address is the link between IdeR, cAMP, cholesterol degradation and phagosomal rupture. IdeR, KstR and KstR2 co-regulate the cholesterol degradation pathway in *M. bovis* [412]. We suggest a similar synergy between IdeR regulation and cholesterol degradation in *Mtb*. Transcription of cholesterol degradation genes in *Mtb* is dependent on the presence of CyA [509]. Regulation of cholesterol degradation by IdeR and cAMP would suggest access to cholesterol is associated to the initial stage of *Mtb* host interaction when the iron pool is oxidized and cAMP is produced to avoid phagosome maturation. Interestingly, EsxA and other pore forming toxins specifically inserts themselves into phosphor lipid (phosphatidylcholine) and cholesterol-containing liposomes [250, 305]. Giant foamy macrophages rich in cholesterol are at the centre of *Mtb* containing granuloma's that turn necrotic [305, 316, 422, 442, 449]. Accumulation of cholesterol was shown to be essential for uptake of *Mtb* by the macrophage [186]. Additionally, cholesterol was shown to increase association of TACO, a coat protein that prevents degradation of *Mycobacteria* upon fusion with lysosomes [186]. We argue that accumulation of cholesterol in macrophages not only increases *Mtb* survival in the phagosome by serving as carbon source but also might assists in phagosomal rupture and possibly in escape from the phagosome.

In summary, in this review we provide an overview for understanding divalent metal homeostasis and their role in regulating three essential virulence strategies of *Mtb*: immune modulation, dormancy and phagosomal rupture. Sensors of environmental and internal cues, including divalent metal availability, form a single regulatory cascade that controls these three virulence strategies. The role of polyphosphate, cAMP and manganese in this cascade requires further investigation.

Funding

This work has been supported by European Union through the SysteMTb project (HEALTH-F4-2010-241587) and the MICROME project, EU Framework Program 7 Collaborative Project [222886–2] and by the Netherlands Consortium for Systems Biology through the "NCSB Stimulus 2013 project: data integratie" project.

Additional Files

Electronic supplementary material can be accessed at the on-line version of Niels Zondervan, **Jesse CJ van Dam**, Peter J Schaap, Vitor AP Martins dos Santos, and Maria Suarez-Diez. "Regulation of Three Virulence Strategies of *Mycobacterium tuberculosis*: A Success Story". In: *International Journal of Molecular Sciences* 2018, 19(2), 347.

Chapter 5

RDF2Graph a tool to recover, understand and validate the ontology of an RDF resource

Adapted from:

Jesse CJ van Dam, Peter J Schaap, Vitor AP Martins dos Santos, and Maria Suarez-Diez. "RDF2Graph a tool to recover, understand and validate the ontology of an RDF resource". In: *Journal of biomedical semantics* 6(1) 2015.

Abstract

Background: Semantic web technologies have a tremendous potential for the integration of heterogeneous data sets. Therefore, an increasing number of widely used biological resources are becoming available in the RDF data model. There are however, no tools available that provide structural overviews of these resources. Such structural overviews are essential to efficiently query these resources and to assess their structural integrity and design, thereby strengthening their use and potential.

Results: Here we present RDF2Graph, a tool that automatically recovers the structure of an RDF resource. The generated overview allows to create complex queries on these resources and to structurally validate newly created resources.

Conclusion: RDF2Graph facilitates the creation of complex queries thereby enabling access to knowledge stored across multiple RDF resources. RDF2-Graph facilitates creation of high quality resources and resource descriptions, which in turn increases usability of the semantic web technologies.

Background

In the life sciences, high-throughput technologies deliver ever-growing amounts of heterogeneous (meta) data at different scales, which are produced, stored and analysed in both structured and semi-structured formats. Systems Biology is an integrative discipline that uses various integration strategies to model and discover properties of biological systems. Integration and analysis of heterogeneous biological data and knowledge require efficient information retrieval and management systems and Semantic Web technologies are designed to meet this challenge [17].

The RDF data model is a mature W3C standard [79, 515] designed for the integrated representation of heterogeneous information from disparate sources and it is proving effective for creating and sharing biological data. RDF is not a data format, but a data model for describing resources in the form of self-descriptive subject, predicate and object triples that can be linked in an RDF-graph. Integration of heterogeneous data from different sources in a single graph relies on using retrievable controlled vocabularies, which is essential to access and analyse integrated data [341]. Once data sources are converted into the semantic Web, SPARQL[22, 396] can be used to query multiple of these resources, simultaneously or consecutively, without further modifying any of them.

Widely used biological resources such as Reactome, ChEBI and UniProt, among other, [50, 119, 219, 254] have been transformed into the RDF data model and the Bio2RDF [55] project has transformed a large set of additional sources, such as the NCBI GenBank files [57], DrugBank [290] and InterPro database [339]. Additionally, there are on-going efforts to develop tools providing results in this data model, such as the Semantic Annotation Platform for Prokaryotes, SAPP, (J. Koehorst, J. van Dam *et al.* personal communication) that provides genome functional annotation in the RDF data model.

These RDF resources can be readily queried with SPARQL. Constructing SPARQL queries requires that the user has a mental representation of the underlying structure of the resource. The structure of a resource is the set of object types and their relationships, i.e. the explicit representation of the predicates linking different classes. This structure represents the set of semantic constraints embedded in the resource. In a biological database containing information on biochemical reactions, genes and their identifiers are linked to proteins; proteins are linked to EC numbers; EC numbers are connected to reactions that involve metabolites as products and substrates. To retrieve information pertaining metabolites and genes, the SPARQL query has to obey the specific network topology linking these types of objects. RDF data sources do not need a predefined scheme so that new data types can be added at any time expanding the underlying structure. If the modifications in the underlying structure generated by this new data are not known, linked information cannot be retrieved. Not having a clear idea of the underlying structure makes querying an RDF resource inefficient, time consuming, or even impossible.

The structure of a resource can be either retrieved through manual queries or it can be provided by the data publishers in the documentation of the resource. This structural information can be encoded using Web Ontology Language (OWL) files [42]. OWL was created as a description logic language and it is intended for automatic reasoning; nevertheless, its axioms can also be used to construct a graphical overview of the described resource [96]. However, the OWL standard does not require all axioms necessary for such reconstruction. Examples of necessary axioms not obligated by this standard are *object all values from* and *data all values from*. In some of the resources created by the Bio2RDF project these axioms (*object all values from* and *data all values from*) are not provided. Furthermore, the ontology generation process is, at best, semi-automatic, time consuming and error-prone. Errors might also accumulate due to the conversion code used to generate the RDF resource, as the triple generating code can contain lexical errors in predicate definition such as typos, inconsistent usage of upper and lower case, or misspelled words, thereby populating a resource containing information on proteins with information on "porteins", which describes proteins associated to transmembrane transport. These errors lead to descriptions of the intended content of the resource rather than of its actual content.

Shape Expressions (ShEx) is a standard to describe, validate and transform RDF data. One of the goals of this standard is to create an easy to read language for the validation of instance data, however, it is still being developed and no final recommendation is yet available [69, 395, 468].

Computational tools able to reconstruct the structure of RDF resources are thus required to i) facilitate query writing and to ii) enable data providers to verify the structural integrity of their resource. To our knowledge, no such tool, able to automatically recover the structure of the resource and the associated multiplicity of the predicates, exist. Semscape [470] is an already existing Cytoscape [463] app that is able to retrieve to some extent the structure of the resource. However, it has limited recovery and simplification capabilities, leading to unreadable hairballs for larger structures. Furthermore, additional statistical information about the classes and links is not retained. Here we present **RDF2Graph**, a tool to automatically recover the structure of an RDF resource and to generate a visualization, ShEx file and/or an OWL ontology thereof. These can be used to write SPARQL queries or to verify (generated) RDF resources.

Implementation

RDF2Graph performs two distinct processes to retrieve the structure of a resource. Initially, there is a recovery of all classes, predicates and *unique type links* together with their associated statistics. In the second stage there is a simplification step to arrive to a neat structural overview. A simplified overview of the complete process is given in Figure 5.1.

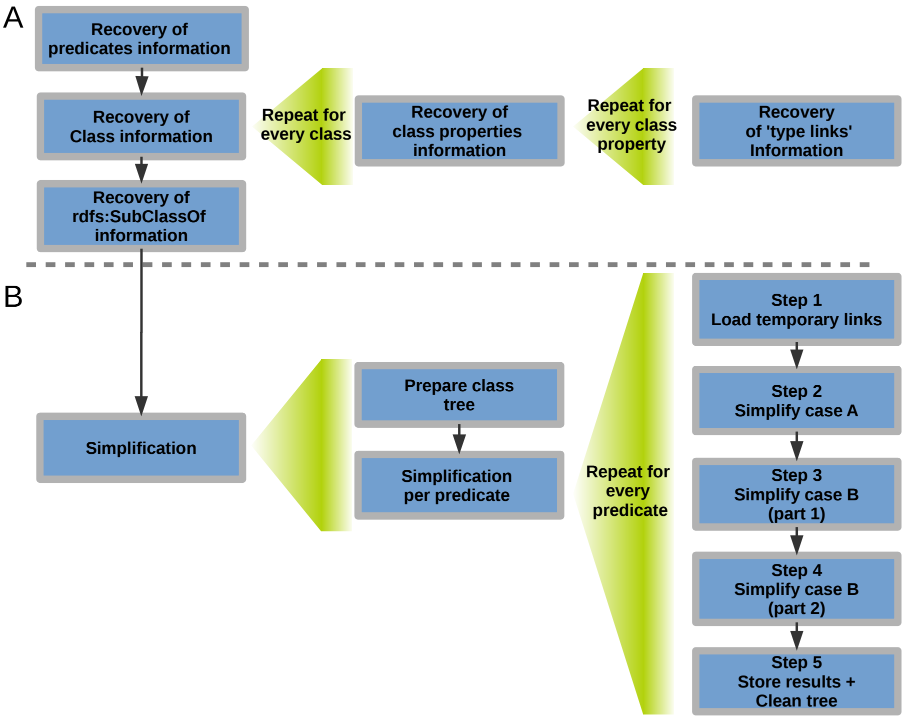

A
Recovery of predicates information
Recovery of Class information
Repeat for every class
Recovery of class properties information
Repeat for every class property
Recovery of 'type links' Information
Recovery of rdfs:SubClassOf information
B
Simplification
Prepare class tree
Simplification per predicate
Repeat for every predicate
Step 1 Load temporary links
Step 2 Simplify case A
Step 3 Simplify case B (part 1)
Step 4 Simplify case B (part 2)
Step 5 Store results + Clean tree

FIGURE 5.1: **Overview of the structure recovery process.** A) Recovery of the information on predicates, classes, class properties, *unique type links* and class hierarchy B) Simplification of the structure leading to a neat visualization by preventing the formation of an unreadable hairball.

A *type link* is defined as a link joining a subject class type to an object class or value data type, via a predicate. A *unique type link* is defined as a unique tuple: *type of subject, predicate, (data)type of object*. For the triple <:BRCA1, :locatedOn, :chromosome17> the *type link* is <:Gene, :locatedOn, :Chromosome>. When considering the full resource, all *type links* <:Gene, :locatedOn, :Chromosome> correspond to the same *unique type link*. In the triple <:Adam :hasSon :Bob> the *type link* is <:Person, :hasSon, :Person>.

The multiplicity of a *unique type link* describes the number of instances connected to each other. The forward multiplicity can be: i) *One-to-one* (also denoted: 1..1) each source instance has exactly one reference to the target; ii) *One-or-many* (1..N) each source instance has one or more references to the target; iii) *Zero-or-one* (0..1) some source instances have at most one reference to the target; iv) *Zero-or-many* (0..N) some source instances have one or even more than one reference to the target. Similarly, for the reverse multiplicity the roles of target and source are inverted. In the previous examples, the forward multiplicity of the *unique type link* <:Gene, :locatedOn, :Chromosome> is (1:1) since each human gene is associated to one and only one chromosome, whereas the reverse multiplicity is (1..N) since a chromosome contains many genes. In the second case <:Person, :hasSon, :Person>, the forward multiplicity is (0..N) since there is no limitation to the number of sons a person may have; in this case the reverse multiplicity is (N=2..1) given that each son has two parents.

The initial recovery process is performed through a series of SPARQL queries on the selected endpoint. Detailed information about the SPARQL queries and the queries themselves are provided in RDF2Graph's documentation. These queries can be adapted to change the introduced limitations and to customise the tool for specific end points. The queries can be limited to reduce the running time since this process can take between a few minutes for a resource with a million of triples, to several days for a resource with 16 billion triples, such as the RDF version of UniProt, as described in the Results section. However the limitation in the number of retrieved triples may lead to incompleteness of the recovered structure, since some *type links* could be missed. This may cause that for some *unique type links* not all *type links* are retrieved, which can cause errors in the calculation of the multiplicities (forward and/or reverse). It may also lead to some *unique type links* not being identified if no *type links* associated to them are found. Therefore, we advise caution when using these limitations.

After the initial recovery of *type links* and *unique type links*, a simplification process follows, in which *type links* with a common parent class for either the subject or object types are merged. These process proceeds in a pairwise manner, so that at each iteration only two *unique type links* sharing either the subject type or the object (data)type are considered. If more than two *unique type links* are present, the first two are merged, and their result is combined with the next one and so on until all have been considered. Therefore, only two *unique type links* at a time are merged. Figure 5.2 represents the cases that need to be considered when analysing two *unique type links*. In principle, other cases involving the "sameAs" relationship could appear, but in our approach, the "subClassOf" relationship also includes the "sameAs" relationship, which reduces all possible cases to the ones represented in Figure 5.2.

This process also allows the identification of *concept classes*. A *concept class* is defined as a class that has no instances and no subclasses with some instances. A typical set of examples of concept classes are all the GO classes in the GO database [28]. This concept is needed to support the exclusion of them in the network view as they have little value for the structural overview and will overcrowd the network visualization.

All classes identified in the recovery process and associated *subclassOf* links are loaded into a memory based directed graph. This class tree is then used in the merging process. During the merging process five steps are executed per retrieved predicate. Step 1 is the initialization; step 2 performs the merging in case A and steps 3 and 4 are used for case B, whereas case C is the combination

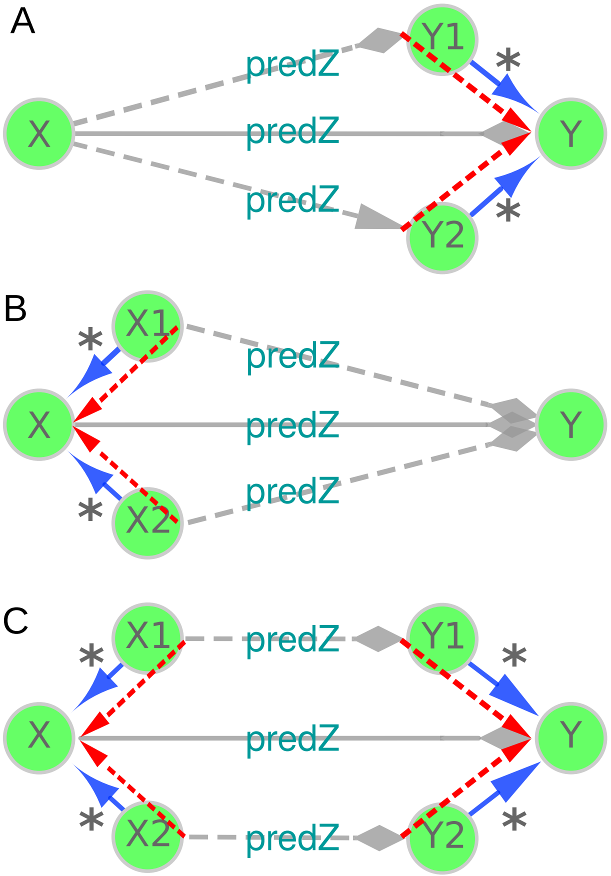


FIGURE 5.2: **Graph simplification by merging of links and classes.** Overview of the possible cases considered in the simplification step. *: Classes X1 and X2 are either equal to X or (indirect) subClassOf X, same applies to Y1, Y2 and Y. **A)** Merging of two links in Class X that link to 2 subclasses of Class Y . **B)** Merging of 2 links in 2 subclasses of class X. **C)** Merging of two links in 2 subclasses of class X that link to 2 subclasses of ClassY.

of cases A and B from Figure 5.2; step 5 is the fictionalization step.

The following definitions of *shared* types and *child of* classes are used. Two types are *shared* if i) both are the same, ii) one is a parent class of the other, or iii) both have a common parent class in the class tree. A *child of class* is defined as follows: Class K is a *a child of* class L if either class K is equal to class L or class K is a (non)direct subclass of class L.

- Step 1: For each *unique type link* found for the predicate currently processed a temporary link is added to the *type of subject*, which links to the *(data) type of object*. In this way a temporary link between both types is defined.
- Step 2: For each class in the class tree all temporary links are copied to the respective parent class(es). Then, occurrences of case A from Figure 5.2 are simplified by performing a search for pairs of temporary links which both point to a *shared* type. If found, the temporary links are merged and replaced by a new temporary link pointing to the common parent class.
- Step 3: This step is executed as a per class recursion breadth first process over the class tree. For each temporary link of the currently processed class the number of direct 'child' classes is counted if they have at least one link pointing to a type that is *a child of* the type pointed by the currently processed temporary link. When this count is one, the currently processed temporary link is removed from the currently processed class.
- Step 4: This step is executed as a per class depth first process over the class tree. Each temporary link pointing to a type that is *a child of* the type pointed by any of the links in the parent classes of the currently processed class are removed.
- Step 5: The remaining temporary links and the newly calculated *unique type links* are stored. The temporary links are cleaned from the class tree to enable the system to process the next predicate.

Results are stored in a local triple store that contains the *unique type links* and their count (number of *type links* associated to them) together with their forward and reverse multiplicities.

To store information for the new concept of *unique type links* we developed a new ontology. Figure 5.3 depicts the elements within this ontology that are related to storage of the *unique type links*. Each *unique type link* links to an object *type* which is either: i) a class; ii) a data type, such as xsd:integer; iii) external, a subject in another resource; or iv) invalid, a subject with no defined type. In each class the *class property* groups the associated *unique type links* per predicate and links them to the rdfs:Property. Additionally, the number of occurrences are stored for each class and predicate.

Results

RDF2Graph recovers the structure of an RDF triplestore endowed with a SPARQL 1.1 endpoint. The results are stored in a local triple store and can be exported to RDF, XGMML, OWL or ShEx files.

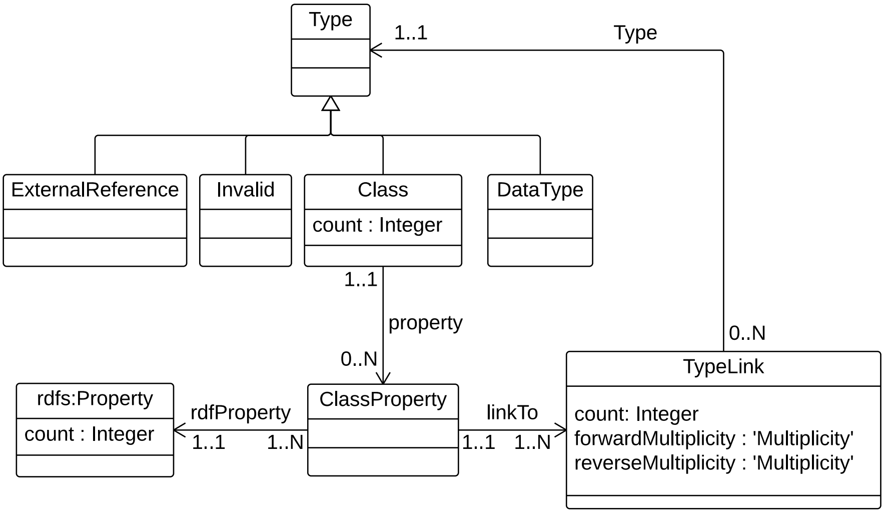


FIGURE 5.3: **RDF2Graph ontology.** A simplified UML diagram of the RDF2Graph ontology.

The RDF export contains the information on the *unique type links*, their count (number of type links associated to them) and their multiplicities (forward and reverse) as stored in the local triple store. The XGMML file provides a graphical format for a network representation that can be opened with tools such as Cytoscape. In the network each node represents a type and edges represents either *unique type links* or *subClassOf* relationships, see Figure 5.4 for additional details. The associated additional information (instance counts, forward/reverse multiplicity and full IRI) are stored as node and edge attributes. The XGMML exporter reports on the *unique type links* for which the multiplicity could not be determined. These correspond to *unique type links* cesing to an invalid subject involved in a set of triples but without a defined type. This can be seen as a measure of the structural integrity of the resource. Additionally, the XGMML exporter reports on i) predicates joined to an invalid subject with properties but no class type definition, and ii) predicates also defined as classes, for instance using CDS (coding sequence) both as a class and a predicate

The OWL representation of the recovered structure contains the following definition and axioms: i) object and data properties, including domain and range definitions, ii) class definition, including the *all values from* restrictions to express *type links* with associated forward and, optionally, inverse cardinalities, and iii) the subclass of definitions from the original recovered resource.

The ShEx compact format contains the shape definitions with the associated class properties, *type links* and forward multiplicities. If a class property

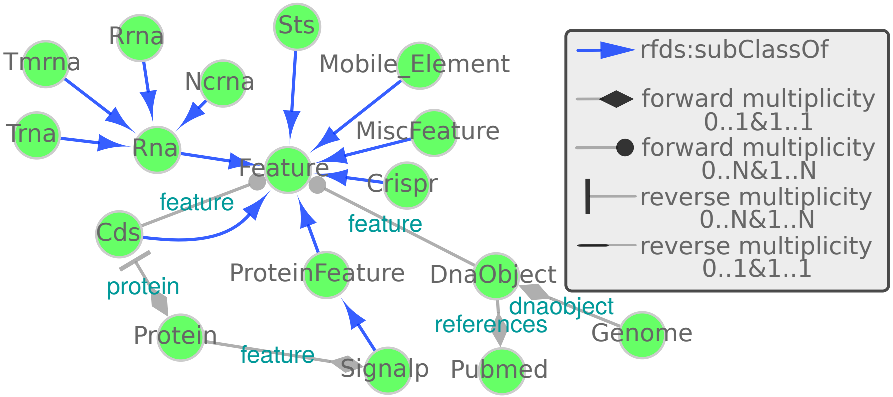


FIGURE 5.4: **SAPP resource.** Network based view generated using RDF2Graph of a resource with genome annotation generated using SAPP. Nodes represent types. Blue edges represent *subClassOf* relationships; Grey edges represent *unique type links*. Arrow heads indicate the multiplicity of the links (see legend in figure). For clarity nodes representing data types have been hidden.

contains multiple *type links* an *or* group is included. Even though ShEx is not yet a fully mature standard, it provides better representation of the *unique type links* than other standards such as OWL. This representation is also more compact and intuitive than the generally used representations of OWL.

Use cases

We successfully applied RDF2Graph to recover the structure of UniProt, EBI Reactome (BioPAX level 3), ChEBI and the RDF dataset generated by SAPP, a semantic, web based, genome annotation tool currently being developed in our group. The statistics of this process are presented in Table 5.1.

The RDF version of the UniProt database contains more than 16 billion triples which is more than one thousand times the number of triples present in the RDF version of Reactome. Therefore there are huge differences in the time required for the recovery process. More than 99% of the CPU time spent in the recovery process is consumed by the SPARQL endpoint.

Given the size of the UniProt resource, we had to impose a limitation on the number of considered *type links* per predicate (100.000). With this limitation the recovery process required more than 740 hours (4 days on 8 cores on a 2.3 GHz computer). On the other hand, the time required to retrieve the structure of Reactome was of only half an hour. This shows that the relationship between the recovery time and the number of triples is non linear. This nonlinearity can be associated to the higher memory requirements associated with a larger resource but also to the intrinsic differences in the structure of the resource, given by the different values of the total number of *unique type links* and the average number of *type links* per class property (see Table 5.1).

TABLE 5.1: **Summary statistics.** Summarizing statistics of the recovery process and recovered structured for UniProt RDF, ChEBI RDF, Reactome BioPAX level 3 and local resource generated by the SAPP tool.

	{UniProt}	**{ChEBI}**	**{Reactome}**	**{SAPP}**
#unique triples in RDF resource	16.313.400.275	425.256.854	14.285.722	359.141
CPU time needed for the recovery	742 h	6,5 h	0,5 h	2 min
#triples in local triplestore				
before simplification	2.507.483	2.854.295	2.411	1410
after simplification	2.504.259	2.848.491	964	912
#classes				
with instances	169	123	45	17
without instances	1.232.947	1.423.143	25	1
# *unique type links*				
before simplification	724	942	254	137
after simplification	302	187	69	78
multiplicity of *unique type links* after simplification				
1..1	53	77	29	46
1..n	11	1	5	2
0..n	104	27	17	4
0..1	128	78	18	26
not determined	6	4	0	0

Even though the UniProt RDF resource has around 40 times more triples than ChEBI RDF, they have a similar number of triples in the local resource. This is due to the high number of *concept classes* and subClassOf relationships that can be found in ChEBI, for example the subClassOf relationship associated with *galactose is an aldohexose*.

The number of triples in the local triple store does not necessarily grow with the number of triples in the resource, since the number of triples in the local triple store is associated with the complexity and the number of classes in the resource, but not with the number of occurrences of each *unique type link*. Table 5.1 shows that the number of triples in the local triple store is roughly equal to twice the number of classes in the resource plus eight times the number of *unique type links*. The number of classes and relationships that can be recovered is limited by the amount of triples that the local triple store can handle. In our case (Jena TDB) that would correspond to an upper limit of roughly 10^7 *unique type links* and classes. So, in practice, the only limitations are given by the restrictions on the SPARQL endpoint imposed by data providers and not by the storage capacity in the local triple store.

Figures 5.4, 5.5, 5.6 and 5.7 provide graphical representations of the retrieved structures for these resources (SAPP, ChEBI UniProt and Reactome) respectively. The nodes in these representations correspond to classes with instances, whereas the edges represent the *unique type links* with determined multiplicity. See additional files for additional output of RDF2Graph regarding these resources (OWL, XGMML, ShEx and the RDF of the local store).

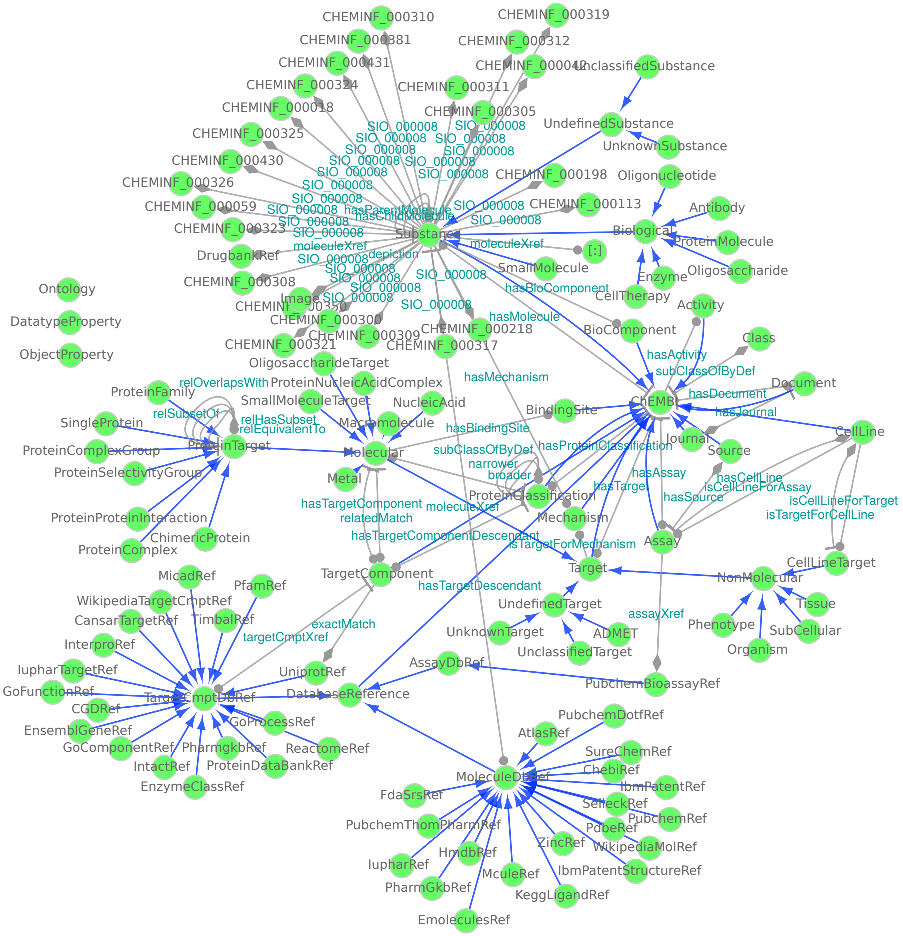


FIGURE 5.5: **ChEBI database.** Network based view generated using RDF2Graph of the ChEBI RDF resource. See figure 5.4 for legend. An interactive XGMML file is included in additional files.

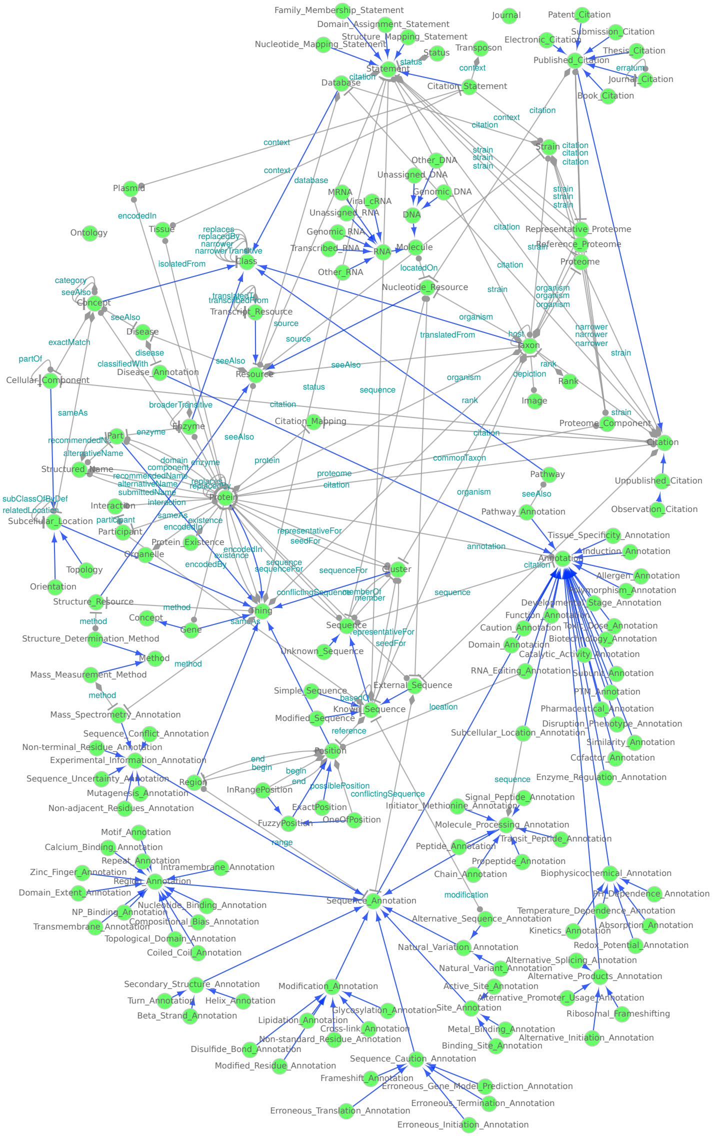
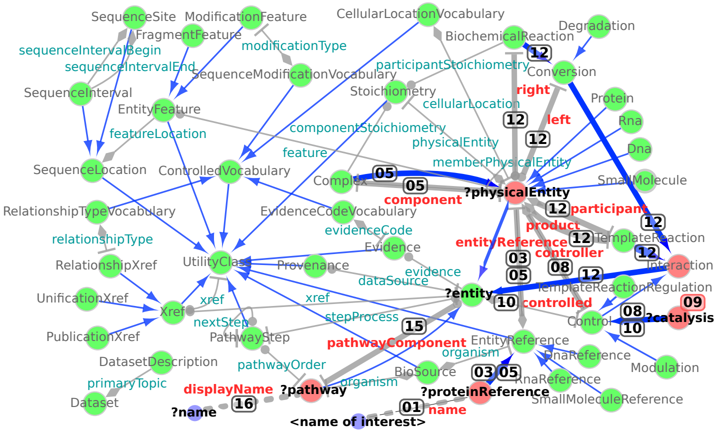


FIGURE 5.6: **UniProt database.** Network based view generated using RDF2Graph of the UniProt RDF resource. See figure 5.4 for legend. An interactive XGMML file is included in additional files.

A



B

```
00 SELECT ?name WHERE { 01 ?proteinRef biopax:name "<name of interest>" .  02
{ 03     ?physicalEntity biopax:entityReference ?proteinRef .
04    } UNION {
05      ?physicalEntity biopax:component/biopax:entityReference ?proteinRef .
06    }
07    {
08      ?catalysis biopax:controller ?physicalEntity .
09      ?catalysis a biopax:Catalysis .
10      ?catalysis biopax:controlled ?entity .
11    }UNION {
12      ?entity biopax:right|biopax:left|biopax:product|biopax:participant
13      ?physicalEntity .
14    }
15    ?pathway biopax:pathwayComponent ?entity .
16    ?pathway biopax:displayName ?name .
17 }
```

FIGURE 5.7: **Reactome database. A)** Network based view generated using RDF2Graph of the Reactome (BioPAX level 3) resource. See figure 5.4 for the legend. An interactive XGMML file is included in additional files. Edges used for the query in B are highlighted. Numbers on highlighted edges correspond to line numbers in B; **B)** SPARQL query to extract the names of all pathways associated to a given gene identifier.

The simplification process reduces the number of *unique type links* from 40% to 80% in the selected resources. Thereby providing a neat structural overview that can be browsed for query building.

There is a small subset of *unique type links* for which the multiplicity could not be determined. These correspond to *unique type links* referencing to an invalid subject involved in a set of triples but without a defined type. These links were identified when using the XGMML exporter.

We used RDF2Graph to incrementally develop and improve SAPP, our semantic annotation tool. For each incremental improvement we recovered the structure and used the graphical overview to assess the integrity of the resource and compare its intended and the actual content. We manually verified every class, associated properties and *type links* to identify and solve an number of issues, such as capitalization errors, predicate naming errors, faulty URI's, broken links, missing attributes and type definitions, unwanted interconnections and faulty multiplicities For example, a broken link will appear as a reference to an external resource where a reference to another class would be expected. A predicate naming error in one of the RDF exporter functions will cause some subjects to be in triples with the "wrong" predicate and will change the multiplicity from *1..1* to *0..1*. Finally, the OWL exporter was used to generate an OWL file requiring little manual curation to complete it.

Similarly we verified the structure of the UniProt RDF resource. The XGMML exported reported 24 errors, most of them associated to subjects with the same missing class definition (see additional files). Additionally we manually compared the provided OWL file with the one created by the RDF2Graph OWL exporter. We detected a set of practical issues such as missing type definitions, references linking the wrong type of objects, incorrect multiplicities and mismatches between the descriptive OWL file and the actual content. These have been reported and will result in an improvement of the quality of this important resource.

The recovered structures and their associated statistics about classes, predicates and *type links* were successfully used to create multiple complex queries. For instance, the retrieved structure of Reactome is depicted in Figure 5.7, panel A. Using this structural information we created the query in Figure 5.7, panel B. This query extracts from Reactome the names of all pathways associated to a specific gene identifier. Through the use of the structural overview we were able to find and follow multiple links from the gene or protein of interest to the associated pathways.

Discussion

RDF resources of biological data are on-going efforts, producing resources that are constantly updated and incorporating additional data sets. Detailed knowledge of the current structure is essential to query and validate these resources and RDF2Graph can be used to understand and improve the quality of an RDF resource. This becomes even more important when the goal is to perform a federated query that spans multiple RDF resources.

Our tool is complementary to existing tools that help create queries such as SPARQL assist [327], Visor [392] iSPARQL [366] and SPARQLGraph [436], these tools are based on local instance or class relationship browsing, or on query suggestion and completion or on a graphical representation of the SPARQL query.

RDF2Graph can be used to inspect instance data (also called *A box*) and semi automatically generate a descriptive OWL ontology. However, it does not check or quantify the quality of the underlying class structure and descriptions (also called *T box*). Nevertheless, there exist several tools such as OntoQA [488], OOPS [393] and OQuaRE [151]) to perform these tasks.

In the provided use cases we performed a manual structural integrity verification. If needed integrity constraint (IC) validation [388] can be used to automatically perform this task. However, for this task an OWL file with all the required axioms is needed. Such an OWL can be generated with RDF2Graph. However, the performance on the generated OWL files upon IC validation implemented by Stardog has not been extensively tested. Additionally ShEx validators, when operational, can be used for this automatic validation, however the output of RDF2Graph ShEx exporter will need to be adapted to the latest definition of ShEx.

So far, RDF2Graph does not support the recovery of contextual links (RDF quads) as they are not supported by the OWL standard although active research is being done to solve this issue [355].

Conclusion

RDF2Graph facilitates the creation of high quality resource descriptions, which in turn improves the quality of the resources themselves. It also facilitates the creation of complex queries, therefore our tool will be helpful for improving the usability of semantic web technologies, which is required for data integration in (computational) biology, systems biology and the emerging field of semantic systems biology.

Availability and requirements

RDF2Graph is distributed under MIT license and it is freely available from `https://github.com/jessevdam/RDF2Graph`. RDF2Graph runs under Linux however, a virtual machine is also distributed with the version described in this manuscript. Furthermore a Galaxy interface is available at `http://semantics.systemsbiology.nl/RDF2Graph/`. The RDF resource size, in this case, is limited to 20.000.000 lines.

Maven 2 [20] is required for installation and the resulting jar can be executed with Java using `bash` or alike. In addition it requires Jena to host the local temporary RDF store and Cytoscape (version 3.x) [463] to generate the network based overview. See the RDF2Graph manual enclosed in the Git repository for more details.

Additional Files

Electronic supplementary material can be accessed at the on-line version of Jesse CJ van Dam, Peter J Schaap, Vitor AP Martins dos Santos, and Maria Suarez-Diez. "RDF2Graph a tool to recover, understand and validate the ontology of an RDF resource". In: *Journal of biomedical semantics* 6(1) 2015.

Authors' contributions

JvD was the primary software programmer of RDF2Graph. JK and MSD participated in the design, testing and documentation of RDF2Graph. JvD and MSD drafted the manuscript. VMdS and PS participated in the conception of the tool and critically revised the manuscript.

All authors read and approved the final manuscript.

Acknowledgements

This work was supported by the European Commission-funded FP7 project INFECT (contract number: 305340).

Chapter 6

Interoperable genome annotation with GBOL, an extendable infrastructure for functional data mining

Adapted from:

Jesse C.J. van Dam*, Jasper J. Koehorst*, Jon Olav Vik, Peter J. Schaap, and Maria Suarez-Diez. "Interoperable genome annotation with GBOL, an extendable infrastructure for functional data mining". In: *bioRxiv preprint* Sep. 5, 2017.

*Equal contributions

Abstract

Background A standard structured format is used by the public sequence databases to present genome annotations. A prerequisite for a direct functional comparison is consistent annotation of the genetic elements with evidence statements. However, the current format provides limited support for data mining, hampering comparative analyses at large scale.

Results The provenance of a genome annotation describes the contextual details and derivation history of the process that resulted in the annotation. To enable interoperability of genome annotations, we have developed the Genome Biology Ontology Language (GBOL) and associated infrastructure (GBOL stack). GBOL is provenance aware and thus provides a consistent representation of functional genome annotations linked to the provenance.

GBOL is modular in design, extendible and linked to existing ontologies. The GBOL stack of supporting tools enforces consistency within and between the GBOL definitions in the ontology (OWL) and the Shape Expressions (ShEx) language describing the graph structure. Modules have been developed to serialize the linked data (RDF) and to generate a plain text format files.

Conclusion The main rationale for applying formalized information models is to improve the exchange of information. GBOL uses and extends current ontologies to provide a formal representation of genomic entities, along with their properties and relations. The deliberate integration of data provenance in the ontology enables review of automatically obtained genome annotations at a large scale. The GBOL stack facilitates consistent usage of the ontology.

Background

Advances in sequencing technologies have turned genomics into a data-rich scientific discipline in which the total assembled and subsequently annotated sequence data doubles every 30 months [158]. To support the growth in data throughput, automated annotation algorithms have become an indispensable supplement to manual annotation [437, 490] and currently, automatic annotations in the UniProt database outnumber manual annotations 100-fold [492].

Functional genome comparison has been used to identify diagnostic markers, to develop effective treatments, and to understand genotype-phenotype associations [12, 118, 153]. The volume and heterogeneity of genome annotation data has created a unique type of big data challenge, namely how to transform computational predicted annotations into actionable knowledge. Tapping into these available resources is only efficiently done by computational means and requires a consistent interlinking of data so that data becomes Findable, Accessible, Interoperable and Reusable (FAIR) [524].

The format for sharing of public genome sequence annotation data has been developed and is maintained by the International Nucleotide Sequence Database Collaboration (INSDC) a long-standing foundational initiative that operates between the DDBJ, EMBL-EBI and NCBI public repositories. However, tradeoffs between simplicity, human readability and representational power left little support for interoperability, i.e. the ability of computer systems to directly make use of information. The */inference* qualifier [240] provides a structured description of evidence that supports feature identification or assignment. Thus, within the standard formats, data provenance of computational annotations could be stored under this optional inference tag but this tag is not designed to be used for contextual, element-wise provenance.

Currently, most annotations rely on computational predictions of structure or function, and the choice of thresholds for confidence scores becomes a key consideration. Tracking the provenance of genome annotations becomes essential for scientific reproducibility and to enable critical reexamination of analyses. However, such meta-analysis is currently very time-consuming. Efficient meta-analysis would require a framework able to accommodate the various types of annotations (e.g. gene prediction, homology, protein domains) directly linked to the supporting statistical evidence. Presently, no machine-readable infrastructure exists to directly query genome annotations linked to the historical and contextual provenance. The World Wide Web consortium provides the Semantic Web and the Resource Description Framework (RDF) data model, supporting these requirements. For RDF, ontologies are essential as they provide consistency in the meaning of data elements and in the relationship between them [226].

In this respect, ontologies already exist for various aspects of biology [43]. The Sequence Ontology (SO) [154] was presented over 12 years ago and was designed as a complete terminology of unambiguous terms related to genetics. However, it was never intended to function as a file format or database schema, and provides no support for linked sets of data attributes. Furthermore, it has limited support for storing based-on provenance except for some experimental codes. FALDO's [68] only purpose is to unambiguously store genetic locations on a sequence. The Synthetic Biology Open Language (SBOL) [182] was successfully designed to describe complete synthetic constructs and the interactions between each of the elements. None of these standards were designed to consistently store feature predictions with evidence provenance and therefore none of these tools provides a complete representation of the genomic information linked to the provenance it is based on.

To meet the requirements and to ensure interoperability of computational predictions, we developed an extendable provenance-centered infrastructure for interoperable genome annotations. The here presented infrastructure consists of two main elements; Firstly, the Genome Biology Ontology Language (GBOL), which directly integrates evidence provenance for the whole dataset and for each included element (dataset- and element- wise provenance). Secondly, the "GBOL stack" of enforcing tools facilitates the consistent usage of ontologies. GBOL is modular in design, extendible and linked to existing ontologies. Empusa has been developed as part of the GBOL stack to ensure consistency within and between ontology (OWL), the API and the Shape Expressions (ShEx) describing the graph structure. This enables the use of SPARQL queries to include contextual details in large scale functional analyses. Modules have been developed to serialize the linked data (RDF) and to generate a plain text format files.

Results

Ontology structure

GBOL is a genome annotation ontology developed for the application of semantic web technologies in genome annotation and mining. As such GBOL provides the means to consistently describe computationally inferred genome annotations of biological objects typically found in a genome sequence annotation data file in the public repositories. Additionally, it can describe the linked data provenance of the extraction process of genetic information from genome sequences.

An overview of the structure of GBOL is shown in Figure 6.1. The ontology contains 251 classes that can be categorized into 6 broad domains (Table 6.1). In GBOL, sequences have features, which in turn have genomic locations on the sequence. The authority of this relationship is derived from the data provenance that captures both the statistical basis of each individual annotation (element-wise provenance) as well as the programs and parameters used for the complete set of sequences under study (dataset-wise provenance). All annotations for a given sequence can be packed into a single entity called a document.

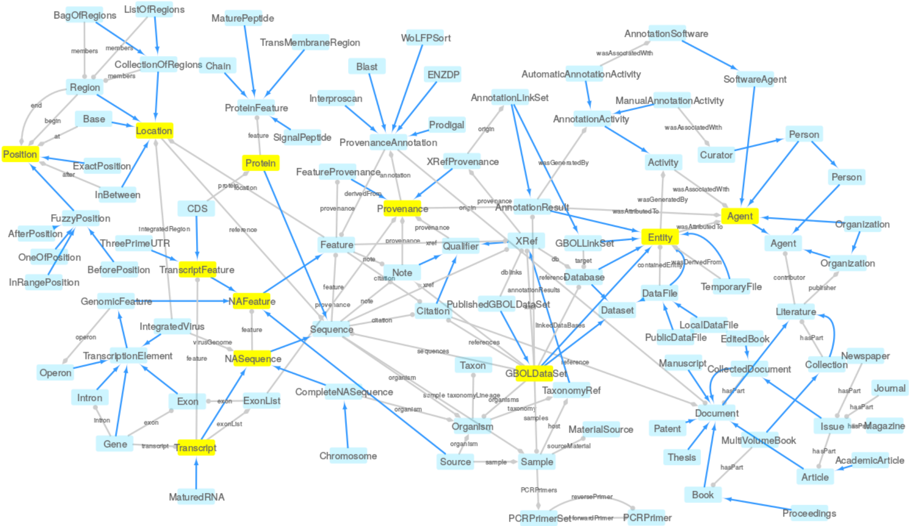


FIGURE 6.1: **The GBOL ontology structure:** Network based view generated using RDF2Graph[125] the GBOL core ontology. Nodes represent types. Blue edges represent *subClassOf* relationships whereas grey edges represent unique type links. A unique type link is defined as a unique tuple: type of subject, predicate, (data)type of object. Arrow heads indicate the forward multiplicity of the unique type links: 0..1 and 1..1 multiplicities are indicated by diamonds; 0..N and 1..N multiplicities are indicated by circles. Neighbourhood of nodes marked in yellow is further expanded in Figures 6.4-6.8

Design principles

GBOL was developed focussing on its function as as file format and as database schema and has the following design principles: modularity, human readability, and annotation. These principles ensure that the ontology can be easily extended [63].

Modularity: The number of classes in the main class tree is kept as small as possible and elements within the data are described with attributes when possible. Furthermore, classes are included in the main class tree only when there are unique properties in a class or in one of the sibling classes. This approach ensures that sub-ontologies can be managed as separate entities within the main ontology and that we can use existing ontologies. As an example the class *RegulationSite* has an attribute *regulatoryClass*, which denotes the type of regulation with a separate set of classes of which all are instances of the regulatory class.

To further simplify the ontology, every attribute is defined as a direct property within the class that links to either a string, an integer, another object or

TABLE 6.1: Overview of domains, classes and properties described by the the GBOL ontology. Note that some properties might be in multiple sub domains.

Sub domain	Classes	Properties	Value sets
Genomic locations	16	17	1
Genes transcripts and features	114	133	17
Document structure	27	107	7
Dataset-wise provenance	22	54	0
Element-wise provenance	5	9	0
BIBO	59	90	2

a class in an enumeration set. For each class in which the attribute is used, an ‘all values from’ axiom is used, with an optional minimal and/or maximal cardinality constraint. The ′all values from′ axiom enforces all referenced objects to be of the expected type, which is not the case with the ′some values of′ axiom and therefore we excluded the use of the ‘some values of’ axiom. This approach is fundamentally different from the principle used in the SO, in which attributes are defined using the ′has quality′ property in combination with the ‘some values of’ axiom that references to a class.

Human Readability: All names within the ontology adhere to a set of basic principles to increase (human) readability of the ontology. All class names represent the underlying biological concept as closely as possible avoiding the use of unreadable numbers. All classes start with uppercase whereas properties start with lowercase. All words are spelled out, and white spaces are left out of the names, instead the next word starts with uppercase. In this way, the class ‘exact position’ becomes ‘ExactPosition’ and the property ‘regulatory class’ becomes ‘regulatoryClass’. Furthermore, where possible, the names are shortened with abbreviations, as long as they remain understandable for a human reader (e.g. XRef instead of CrossReference).

Annotation: All classes and terms within the ontology are annotated with a short definition; an optional comment with additional usage information; an optional editorial comment relating to the development of the ontology itself; an optional *ddbj* label indicating the presence in the GenBank standard; and an optional SKOS [338] exact match to relate classes to terms in existing ontologies.

The GBOL infrastructure

An infrastructure enabling interoperable genome annotations integrated with provenance requires the following characteristics: i) An OWL [18] encoded definition of an ontology. ii) An infrastructure to enhance and simplify its usage, consisting of an interface (API) that allows to use Java and R. iii) A file format that can be obtained from serializing the linked data (RDF) using a lightweight Linked Data format (JSON-LD) [471] which is subsequently serialized as YAML [56]. This format mimics the layout of the current format for sharing of public genome sequence annotation data, but has integrated support to add additional information. iv) A ShEx definition for data conformance validation to enhance data consistency [395]. And v) a tool to convert existing GenBank and EMBL format files into the GBOL format.

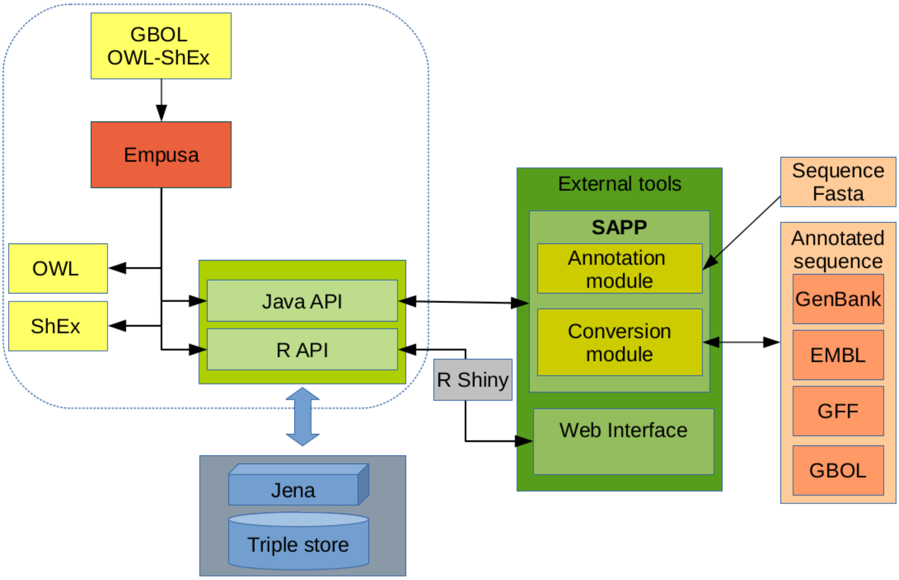


FIGURE 6.2: **Schematic of an interoperable provenance centered genome annotation pipeline.** The GBOL stack (dashed box) provides the Genome Biology Ontology Language (GBOL) (Yellow) and associated infrastructure to keep it consistent and extendable (Empusa). The SAPP module functions as an interface for (standardly used) genome annotation tools. Using the Java API, SAPP retrieves raw genome data from the triple store, runs genome annotation tools in batch and uses the GBOL ontology to automatically store their predictions and associated data provenance directly as RDF triples in the triple store database (Blue). Stored predicted functional annotations, data provenance and linked meta-data can be queried within Java and R with SPARQL and by using a web interface (Green). Parsers have been developed for conversion of annotation files in standardly used formats (Orange).

GBOL data can be stored in any of the linked data formats (RDF), such as Turtle. The generated API can be used to access the genomic information encoded within the GBOL format, which includes a data consistency validation. The API directly reads from and directly modifies the RDF data structure upon usage of any of the data model functions. This enables the usage of SPARQL within the client code, which can run a SPARQL query and directly use the resulting objects nodes in the API. Moreover, the RDF data can be structured into a tree with the JSON-LD framing API into JSON-LD, which, in turn, can be further serialized as YAML resulting in a human readable format for sharing of public genome sequence annotation data. By addition of standard annotation tools, the GBOL stack can be at the core of a provenance-centered genome annotation framework (Figure 6.2).

Embedding with other ontologies

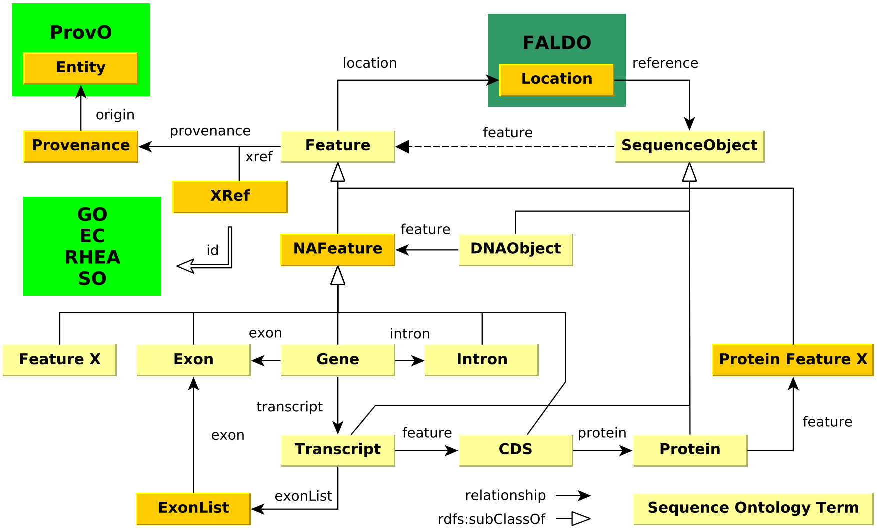


FIGURE 6.3: Embedding of the GBOL ontology with already existing ontologies. FALDO, ProvO, GO, EC, RHEA and SO are existing ontologies. Classes are in yellow and an explanation is provided in the main text.

GBOL is embedded in the corpus of currently developed web technologies and when possible we have integrated existing ontologies such as: FALDO [68], PROV-O [291], SO [154], SBOL [182], BIBO [195], WikiData [340], FOAF [81], Gene ontology (GO) [28] and the Evidence ontology [107] as depicted in Figure 6.3. Annotation of genomic location is inspired by FALDO ontology, although several elements had to be modified. The PROV-O ontology was used and extended to store data provenance. Whenever applicable, we added a cross-link to exact matching terms within the FALDO, SO and SBOL ontologies. Identification of persons and institutions is done through the FOAF ontology and BIBO is used to identify publications.

GBOL does not represent a vocabulary to describe genetic, molecular or cellular functions. Instead, terms can be cross-referenced to the many vocabularies that provide functional descriptions to the (products of) genetic elements, such as Gene Ontology, Enzyme commission (EC) numbers, and the ChEBI and RHEA databases [11, 132], among others.

Key GBOL classes

Common elements in genome annotations include different classes of DNA molecules such as chromosomes, plasmids and contigs, genes, transcripts, exons, introns, proteins, protein domains and functional annotations. The following sections summarize the key classes of the ontology. An extensive description for each element can be found in the documentation available at `http://gbol.life/0.1/`.

Genomic locations: Genomic locations of all features in GBOL is captured with the *Location*, *Position* and *StrandPosition* classes, which are inspired by the FALDO ontology and represented in Figure 6.4. The *Location* and its subclasses together with the *StrandPosition* define an interval on the Sequence, whereas *Position* defines a single position in a sequence. A location can be either: i) A region which has begin and end positions; ii) A collection of regions (ordered or unordered); iii) A single base at a given position; or iv) an *InBetween* location denoting a location between two bases after the base of which the position is given. Each region, base and in-between location can be defined to be located on the forward, reverse or both strands, although no strand should be specified if the sequence is a single stranded DNA sequence or a protein sequence. It should be noted that elements of a collection of regions can be located on different sequences. This can be used to encode cases in which an otherwise indistinguishable genetic element is located on multiple chromosomes.

Exactly known positions can be indicated using the *ExactPosition* class containing the *position* property. Otherwise a not exactly known position, also called fuzzy position, can be indicated using either the *BeforePosition* class containing the *position* property, the *AfterPosition* class containing the *position* property, the *InRangePosition* class containing the *beginPosition* and *endPosition* properties or the *OneOfPosition* class containing multiple *position* properties.

Genes, transcripts and other commonly encountered genomic features: GBOL has a consistent model for storing genes, exons, (alternatively spliced) transcripts, coding sequences and proteins. Central to this model is the *Sequence* class that can have multiple annotations represented in the *Feature* class. An overview is provided in Figure 6.5.

In GBOL a sequence can be specified as a nucleic acid (NA) or a protein sequence. The sequence is attached to the *Sequence* class via the *sequence* property, provided in the DNA, RNA or protein encoding standard. NA-sequences can represent transcripts or other elements such as chromosomes, plasmids, scaffolds, contigs or reads. No distinction is made between DNA and RNA and the *strandType* denotes that it is either a double or single stranded DNA or

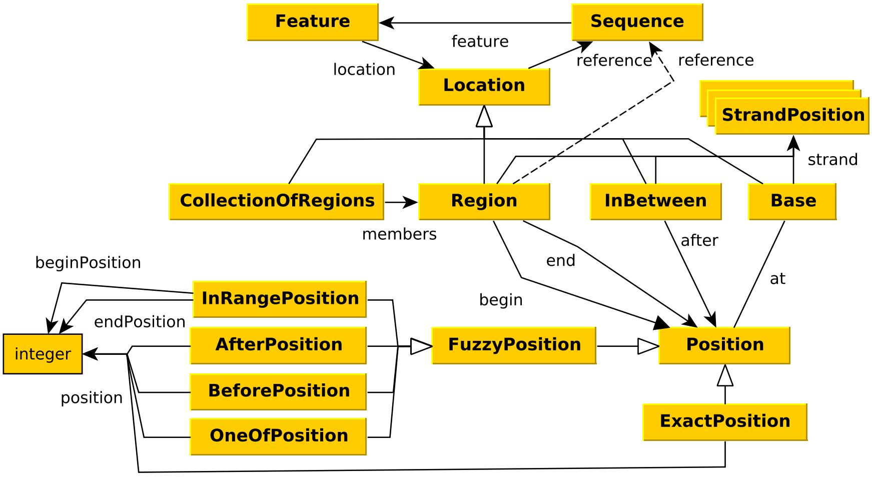


FIGURE 6.4: Graphical view of the GBOL ontology for genomic locations. An explanation of the classes is provided in the main text.

RNA. As indicated in Figure 6.5 the type of sequence determines the features it might be associated to (*ProteinFeature*, *NAFeature* or *TranscriptFeature*),

Typically, each *GBOLDocument* contains one or more *NASequences* (e.g. *Chromosome*, *Contig*, *mRNA*), which can have multiple features including all gene, exon, intron, sequence variations, and structural, regulatory and repeat annotations. Each gene is linked to its associated exons, introns and transcripts. Due to alternative splicing a gene can have multiple transcripts. Each transcript has its own unique list of exons, which is linked through the *exonList* and associated *exonList* class to all associated exons. A transcript can be either a mRNA, ncRNA, rRNA, tmRNA, tRNA, precursor RNA or a miscellaneous RNA. The type of transcript determines the associated features: mRNA transcripts can have features linked to coding sequence (CDS), 5'-UTR, 3'- UTR and poly A tail.

The mRNA translation table is defined with the *translTable* property from the parent sequence. The association between CDS and the encoded protein is preserved and information about the translation is stored if it is different from the default translation (for example, use of alternative stop codons).

Each protein has a unique IRI (`http://gbol.life/0.1/protein/<SHA-384>`) based on the SHA-384 hash of its sequence. This makes it possible to combine protein information from heterogeneous sources, as a protein can be associated to several CDS features. All information related to the protein which is unique to the genome (such as location) should be stored in the CDS feature. Protein annotation features may include, among other, conserved regions, protein domains, binding sites, 3D structure, signal peptides,

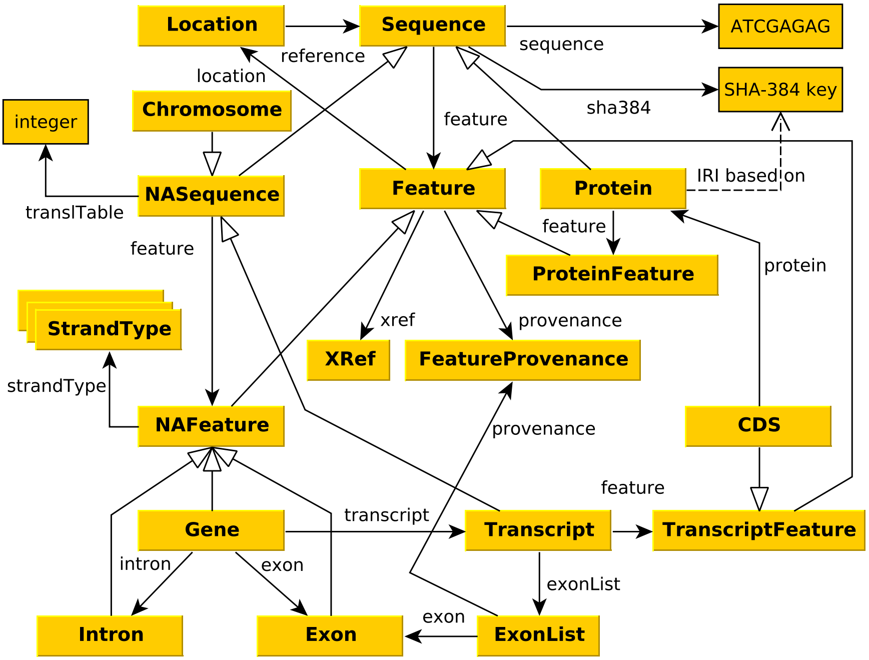


FIGURE 6.5: Graphical view of the GBOL ontology for genes, transcripts and other commonly encountered genomic features. An explanation of the classes is provided in the main text

transmembrane regions, and immunoglobulin regions. Operons can be defined with the *Operon* feature, to which other genomic features, such as genes, can be associated. Additionally, viral genome integration can be denoted using the *IntegratedVirus* feature.

Provenance related classes

Three types of provenance can be distinguished. Metadata refers to the owners of the samples, the biological origin, culture conditions etc. Dataset- and element- wise provenance pertain to the annotation process. All data within a single data collection stored in GBOL is based on the *GBOLDataSet*, which holds among other, references to all included samples, sequences, organisms, annotation results and linked databases. An overview of the document structure is given in Figure 6.6.

A sequence originates from a sample and samples are related to one or multiple organisms. The *sample* property which links to the *Sample* class describes where, when, how, by whom and from what the sample was collected.

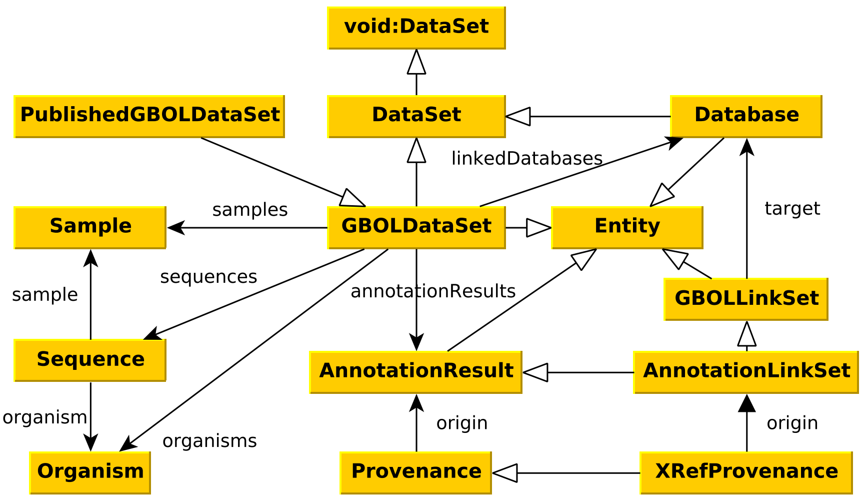


FIGURE 6.6: Graphical view of the GBOL Document structure. An explanation of the classes is provided in the main text

The fields follow the GenBank format. The *organism* property describes the taxonomic reference, its scientific name and its taxonomic lineage.

All annotations made within the *GBOLDataSet* have associated provenance and should originate from one of the listed annotation results, so that correspondence with originating databases is preserved. The *Database* and the *GBOLDataSet* classes are both sub classed from the void ontology, *Dataset* class contains a general description, including among other title, description, comment, license, version, data download address, SPARQL endpoint URI, and URL encoding.

Dataset-wise provenance: Storage of the dataset-wise provenance is based on the PROV-O ontology in which the *Entity*, *Agent* and *Activity* classes are central. An activity can use and generate entities, which are executed (*wasAssociatedWith*) by an agent. As a result, an entity can be attributed to an agent. The GBOLDataset, AnnotationResult, *GBOLLinkSet* and *Database* classes (indicated in Figure 6.6 and 6.7) are subclasses from the PROV-O ontology *Entity* class, so that for each of these objects provenance on how, when and by whom they were created can be associated.

In GBOL an *Entity* is either a file or an annotation result. The annotation result is a set of triples contained within a GBOL document, whereas a file represents a physical file either on a computer or network. An *agent* can either be a curator, person, organization or annotation software. For the annotation software a version and code repository with associated commit identifier is included to enable univocal identification. For a curator, an ORCID [214]

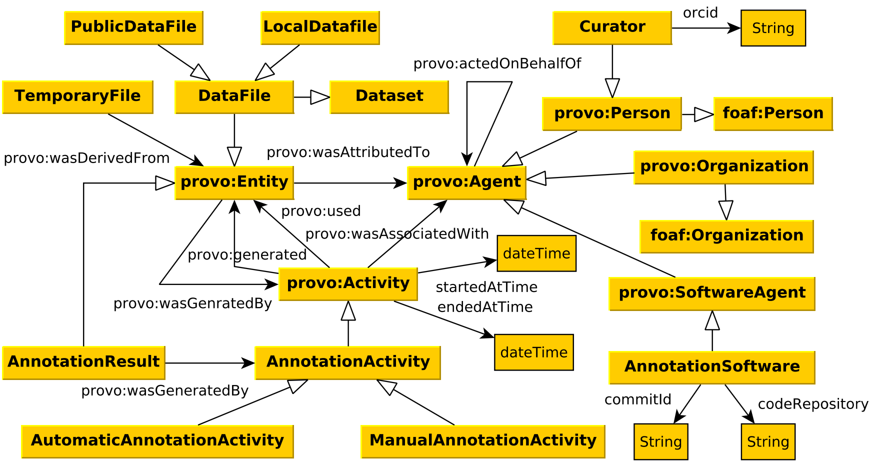


FIGURE 6.7: Graphical view of the GBOL Dataset-wise provenance. An explanation of the classes is provided in the main text

must be specified so that each curator can be uniquely identified together with his/her organization. Both *Person* and *Organization* are sub-classed from the FOAF ontology to include additional information such as name and email address.

Within GBOL, each activity is an annotation activity, which can be either an automatic process or a manual curation activity, with a start and end time. An automatic annotation must be associated with a software agent and the set of parameters used must be specified including the corresponding input and/or output files. Finally, manual curation must be associated with a curator.

Element-wise provenance and qualifiers: In addition to the dataset-wise provenance, GBOL is able to capture an additional layer of element-wise provenance, as the provenance of all the annotation in GBOL is captured per property per feature with the *FeatureProvenance*, as shown in Figure 6.8. For properties that could have items from multiple sources, we have defined the *Qualifiers*, each with its own associated provenance. A qualifier can either be a *citation*, *note* or cross reference (indicated by *xref*). A citation can hold a reference to literature encoded with the BIBO ontology.

Annotations are linked to the provenance object either through the *provenance* property of the qualifiers or the *onProperty* property of the *Provenance* feature. The provenance object links to both the dataset-wise provenance and the element-wise provenance. The *origin* links the provenance with the dataset-wise provenance (*AnnotationResult*), which includes among other the

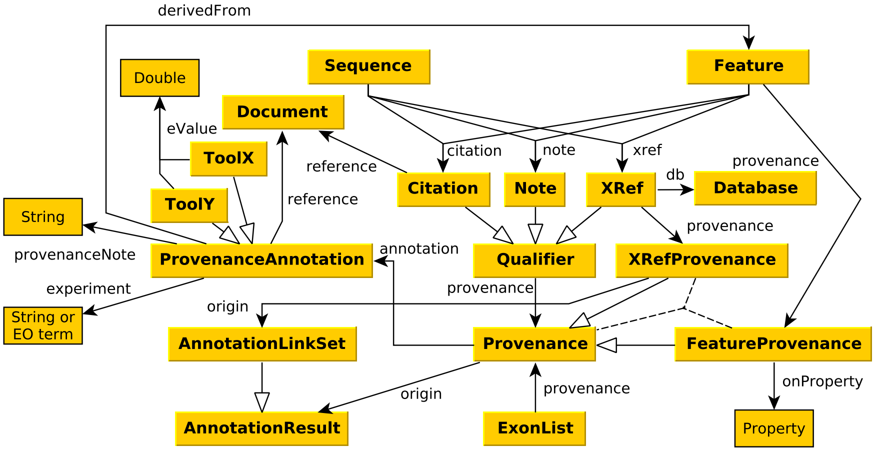


FIGURE 6.8: Graphical view of the GBOL element-wise provenance. An explanation of the classes is provided in the main text

creation time, identity of the creating agent and the used parameters, as previously mentioned. The *annotation* links to the element-wise provenance (*ProvenanceAnnotation*), which includes: A free text note to describe the annotation; A list of references supporting the note; An experimental code, preferably from the Evidence Ontology to qualify the evidence supporting the conclusion; An optional *derivedFrom* that links to other features on which it is based.

Finally, each annotation tool generates its own evidence statements, often embedded in a statistical framework, characteristic of the algorithmic approach taken, such as p-values, bit scores, matching regions or any other scoring system. To store tool specific confidence scores, a subclass of the *ProvenanceAnnotation* class can be created. Some example classes include *BLAST*, *HMM* and *SignalP* associated with the output of corresponding tools [90, 389, 400] However, these classes are not part of the GBOL ontology itself.

Empusa

During the development of the standard, difficulties were encountered in managing the large set of properties and structures in the OWL and ShEx definitions and the API needed to encode the annotation information in conjunction with the associated provenance. Moreover, Analyses of various public repositories have shown that inconsistent, non-enforced usage of ontologies leads to mismatches between the descriptive OWL file and the actual content [125]. In order to shorten the development cycle and to maintain consistency within and between the OWL and ShEx definitions and the API, a standalone tool was developed named Empusa. The input definition of Empusa is a combination between OWL and a simplified version of ShEx, which can be edited within Protégé [347]. The classes are defined in OWL, whereas the properties are defined in each class under the annotation property 'propertyDefinitions' encoded within a simplified format of the ShEx standard. Additionally predefined value sets (for example all regulatory types) can be defined by adding a subclass to the EnumeratedValueClass. Each subclass of the value set is represented as one element within the value set. As standalone tool, Empusa can automatically and consistently generate an OWL and a ShEx definition, ontology documentation in markdown, an API, a JSON-LD framing file and a visualization. Empusa uses parts of the RDF2Graph tool [125] to generate a representation that can be subsequently used to generate a visualization within Cytoscape [444]. This allows users to browse the complete ontology intuitively.

Discussion

Comparative genome analysis is essential to understand the mechanisms underlying evolution and adaptation. Ideally, comparative genomics should be performed at the functional level, as this is highly scalable and more resistant to phylogenetic distances [271]. However, as functional annotation is performed in a non consistent manner the current practical level of interoperability is at the sequence level. Many tools exists to obtain orthologous clusters which are shaped by a generalised acceptance threshold for similarity and alignment length which is a trade-off between sensitivity and false discovery [176, 298]. At large scales these analysis are hampered by the high computational cost for finding bi-directional best matches. We have shown [271] that functional comparison, based on consistently annotated protein domains, provides a fast, efficient and scalable alternative .

The prerequisite of a direct comparative functional analysis is consistent annotation of the genetic elements with evidence statements. Recording the provenance allows class-specific cut-offs for each individual annotation. Element-wise provenance enhances the re-usability of the annotations, and allows the development of methods to combine evidence statements, often derived from complex statistical frameworks, into confidence statements. Element-wise provenance also enables a quick re-evaluation of evidence, for instance by using a tunable cut-off score.

/beginsloppypar GBOL has been developed to explore available genome sequences using the mining possibilities of linked data. As a result, GBOL has evolved to consistently capture annotation data generated by the Semantic Annotation Platform with Provenance (SAPP), available at `http://semantics.systemsbiology.nl`. Previous versions of the GBOL ontology have been used to compare 432 *Pseudomonas* strains through integration of genomic, functional, metabolic and expression data [272]. Here GBOL was essential to capture, store and interlink the genomic and functional annotation data. Strikingly, over 432 *Pseudomonas* strains, consistent *de-novo* annotation yielded 838 additional GO-terms and 146 additional protein domains which would not have been identified using the original gene predictions. In addition to determining the functional pan- and core genome of a species, comparative genomics also enables the investigation of genotype-phenotype associations. In [258] we consistently functionally annotated and compared 80 publicly available mycoplasma genomes. The resulting semantic framework allowed us to efficiently query for functional differentiation of various mycoplasma species in relation to host specificity and phylogenetic distance. /endsloppypar

Consistent functional annotation within a semantic framework requires a standardised ontology for the annotated elements and the associated based-on provenance. Linked data ensures that queries can be performed, mining multiple sequences at once, thereby providing a scalable alternative for large scale genome comparisons. The GBOL stack provides the ontology and corresponding API that enables the incorporation of functional annotation and provenance reducing complexity and is the outcome of efforts in a number of studies related to functional comparative genomics. Currently the GBOL stack is being used in various collaborative projects to handle genomic data of organisms across all domains of life [141, 157, 239, 348].

GBOL has been primarily designed to handle genomic annotation. However, it has been designed in a modular and extensible manner so that in the future it can be extended to host other omics data types as proteomics and transcriptomics. The modular design of GBOL ensures that other ontologies can be incorporated and managed as separate entities. For instance, the majority of the feature and sequence classes within GBOL can be connected with those from the Sequence Ontology and are therefore linked with the *skos:exactMatch* predicate. The major difference between GBOL and SO is that SO has been defined as vocabulary of terms related to genetic elements, whereas the GBOL classes have been designed to describe genetic annotation and elements located on a sequence and is inspired on the principles of the GenBank format. However, still a number of features in the SO are not currently available in GBOL and future work should focus on including them. Another possible extension would be to link to other Minimum Information Standards like MIGS and extensions thereof (MIMARKS, MIxS) [168, 534] and cross domain experiment reporting standards like ISA-tab [415]. Other possible extensions relate to the development of the sub-ontologies GBOL links to. For instance, BIBO is used to store information on literature references, however the OWL ontology file of BIBO has to be further improved, as it does not specify to which classes all of the properties should belong. Therefore we have chosen to include a less consistent representation of the properties by adding all properties to the root class *bibo:Document*.

Empusa, a core part of the GBOL stack, ensures the correct usage of the ontology through the provided R and Java API. We have ensured that Empusa can be used independently of GBOL (documentation available at `http://`

`gbol.life`) and therefore can be used to develop new ontologies combined with an automatically generated API and documentation. This reduces the complexity and time to extend and develop ontologies with corresponding API's and ensures consistent and correct usage of a defined ontology.

Conclusions

Large scale analysis of heterogeneous biological data is hampered by lack of interoperability. To improve the exchange of information formalized information models are required. GBOL provides a formal representation of genomic entities, their properties and relations. The GBOL Stack provides a framework to enforce consistent and correct usage of GBOL. The semantic basis and the integration of provenance enables FAIR genome annotations, thereby unlocking the potential of functional genome annotation data.

Methods

The GBOL ontology is OWL encoded and a ShEx schema is provided. All supporting software (Java and R API, Empusa) are written in Java with Gradle as build system. We use Jena [21] for handling and loading the RDF data into a triple store. Protégé was used for editing the ontology[347].

Storage of the genomic location is inspired by FALDO, although several elements had to be modified e.g. to account for features that start and end on different sequences. Differences include: i) *StrandPosition* is not subclassed from *Position*. Instead, an additional property is added to the region, *base* and *InBetween* location, this is done because these location object types can have both a strand position and an index position on the sequence. ii) The *reference* property is not part of a Position, but of a Location, because a location that starts on one sequence and ends on another sequence is an undefined sequence. iii) The *BaseLocation* and the *InBetweenLocation* classes have been added to the ontology. iv) The *BaseLocation*, *InBetweenLocation*, *CollectionOfRegions* and *Region* are children of the *Location* class, such that the rest of the ontology can incorporate these classes. v) The *before* and *after* positions have been explicitly defined to include their semantics. vi) The classes sub-classed from *FuzzyPosition* have an integer to denote the position and do not point to another position object, which could allow for arbitrary complex location denotations. vii) The N- and C-terminal positions have been removed and all indexes are counted from the N-terminal side. Counting from the C-terminal side can be calculated based on the sequence length. vii) The reflective properties *beginOf* and *endOf* have been removed, because a position can also be referenced by the added base location. For consistency we have redefined all FALDO elements within our own namespace.

Cross-links to exact matching terms from other ontologies (such as FALDO, SO and SBOL) where added using skos:exactMatch. Additionally, several properties within the ontology point to existing ontologies, for instance: i)

The *signalTarget* property of SignalPeptide, the *modificationFunction* of *ModifiedResidue* and the *organelle* of *Sample* are interlinked with GO terms. ii) The *experiment* property of ProvenanceAnnotation, which denotes upon which evidence the annotation is based on, should point, where possible, to a term within the Evidence Ontology. iii) The *residue* property of *ModifiedResidue* must point to a term within the Protein Modification Ontology [342]. iv) GBOL includes the GO terms for *tissueType* of the Sample class and points, when possible, to a term within the BRENDA Tissue and Enzyme Source Ontology [435].

The source file of the ontology encoded in the Empusa and associated generated OWL definition, ShEx schema and visualization for Cytoscape available at `http://www.gitlab.com/GBOL` under the MIT license. The generated Java and R API are available at `https://gitlab.com/gbol/GBOLapi` and `https://gitlab.com/gbol/RGBOLApi` under the MIT license. The conversion module, which is part of SAPP, is available at `http://www.gitlab.com/SAPP/conversion` under the MIT license. The supporting Empusa code generator is available at `http://www.gitlab.com/Empusa` under the MIT license. All projects are coded in Java and are based on the Gradle build system. All terms are resolvable and can be browsed for at the associated website `http://gbol.life/0.1/`.

Acknowledgements

We thank Benoit Carreres for helpful design discussions.

Funding

This work has received funding from the Research Council of Norway, No. 248792 (DigiSal) and from the European Union FP7 and H2020 under grant agreements No. 305340 (INFECT), No. 635536 (EmPowerPutida) and No. 634940 (MycoSynVac).

Chapter 7

SAPP: functional genome annotation and analysis through a semantic framework using FAIR principles

Adapted from:

Jasper J. Koehorst, **Jesse C.J. van Dam**, Edoardo Saccenti, Vitor A.P. Martins dos Santos, Maria Suarez-Diez and Peter J. Schaap. "SAPP: functional genome annotation and analysis through a semantic framework using FAIR principles". In: *Bioinformatics* 34(8), 2018, 1401–1403

Abstract

Summary: To unlock the full potential of genome data and to enhance data interoperability and reusability of genome annotations we have developed SAPP, a Semantic Annotation Platform with Provenance. SAPP is designed as an infrastructure supporting FAIR *de novo* computational genomics but can also be used to process and analyse existing genome annotations. SAPP automatically predicts, tracks and stores structural and functional annotations and associated dataset- and element-wise provenance in a Linked Data format, thereby enabling information mining and retrieval with Semantic Web technologies. This greatly reduces the administrative burden of handling multiple analysis tools and versions thereof and facilitates multi-level large scale comparative analysis. **Availability:** SAPP is written in Java and freely available at `https://gitlab.com/sapp` and runs on Unix-like operating systems. The documentation, examples and a tutorial are available at `https://sapp.gitlab.io`.

Introduction

Managing the genomic data deluge puts specific emphasis on the ability of machines to automatically find and use the data. To meet this demand and to extract maximum benefit from research investments, digital objects should be Findable, Accessible, Interoperable and Reusable (i.e. FAIR) [524].

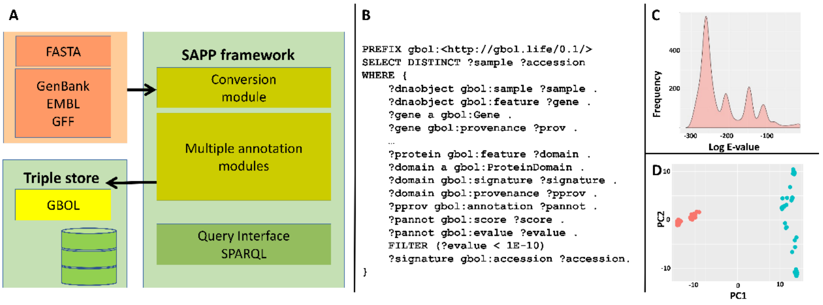
Genome annotation data is usually findable and accessible through public repositories in which the data is linked to metadata providing detailed descriptions of the data acquisition and generation process. Interoperability reflects the potential for seamless integration of data from independent sources. Currently, genome comparisons usually involve a laborious process of data retrieval, modification and standardization (canonicalization). Reusability requires rich metadata with provenance for each annotation. Current standard formats (GenBank, EMBL or GFF3) retain the output of the prediction tools (for example for gene identification) but only when they score better than a predefined, often pragmatic, prediction threshold. Detailed information of the actual prediction scores is lost. This hampers critical re-examination of the results.

Because existing genome annotation data is hard to be made FAIR and managing of FAIR genome annotation data requires a considerable administrative load, we developed SAPP, a semantic framework for large scale comparative functional genomics studies. SAPP can automatically annotate genome sequences using standard tools. The unique characteristic of SAPP is that the annotation results and their provenance are stored in a Linked Data format, thus enabling the deployment of mining capabilities of the Semantic Web. As the automatic annotations are incorporated into a dynamic framework, SAPP supports periodic querying, comparison and linking of diverse annotation sources, resulting in up-to-date genome annotations. By interrogating metadata as part of a digital annotation object, annotation data becomes interoperable as the extraction procedure requires no additional standardization process.

Implementation

SAPP accepts annotated and non-annotated sequence files which are converted into an RDF data structure using the GBOL ontology [124]. Within SAPP, structural and functional annotation is performed using add-on modules incorporating existing standard annotation tools such as Prodigal and Augustus [235, 472]. Modules for tRNA, tmRNA, rRNAs, protein domain and CRISPR repeats annotation are also available. New modules can be added. Annotation data and metadata are stored in a compressed graph database [167], as shown in Fig.7.1A.

Genome annotations can be exported to standard formats. All data can be directly queried and compared using the SPARQL endpoint or via the GBOL API (Java/R). Complex queries can be performed on multiple genomes while simultaneously taking meta-data into account. A SPARQL query example is provided in Fig. 7.1B. Examples to query SAPP from R, Java or Python, a tutorial and a list of publications in which SAPP was used can be found at `http://sapp.gitlab.io`.



```
PREFIX gbol:<http://gbol.life/0.1/>
SELECT DISTINCT ?sample ?accession
WHERE {
    ?dnaobject gbol:sample ?sample .
    ?dnaobject gbol:feature ?gene .
    ?gene a gbol:Gene .
    ?gene gbol:provenance ?prov .
    ...
    ?protein gbol:feature ?domain .
    ?domain a gbol:ProteinDomain .
    ?domain gbol:signature ?signature .
    ?domain gbol:provenance ?pprov .
    ?pprov gbol:annotation ?pannot .
    ?pannot gbol:score ?score .
    ?pannot gbol:evalue ?evalue .
    FILTER (?evalue < 1E-10)
    ?signature gbol:accession ?accession.
}
```

FIGURE 7.1: **A:** The conversion module imports genome sequences in common formats. Annotation modules perform common tasks such as gene, tRNA, protein and protein domain annotation. Results are stored as Linked Data and consistency is ensured by the GBOL stack. **B:** SPARQL query to retrieve the E-value score of the instances of the protein domain PF00465 across multiple bacterial genomes. **C**: Distribution of E-values for protein domain PF00465 across multiple bacterial genomes: note the multimodality of the distribution. **D:** Principal component analysis of functional similarities of 100 bacterial genomes from the *Streptococcus* (blue) and the *Staphylococcus* (orange) genera. PC1 and PC2 account for 51.4% and 10.1% of the variance in the dataset respectively.

Results and Discussion

Reproducible computational research requires a management system that links data with data provenance. Interoperability requires a strictly defined ontology. Using and sharing Linked Data based on controlled vocabularies and ontologies ensures the interoperability and reusability of the data. SAPP functionalities are unique since none of the existing *de novo* annotation pipelines implement Semantic Web technologies. SAPP generated data fulfil the applicable requirements for data FAIRness proposed by [524].

For input and output, these tools interact directly with the database thereby forcing automatic linkage of data and provenance. In this way there is no need to work with predefined thresholds on the parameters controlling the annotation output. SAPP uses a controlled vocabulary to describe genome annotations. Consistency is ensured through the GBOL Stack [124].

The GBOL ontology enables consistent genome annotation while integrating dataset-wise and element-wise provenance. The element-wise provenance is the statistical basis or score of each individual annotation, whereas the dataset-wise provenance refers to the programs, versions thereof and parameters used for the complete annotation of the (set of) sequences under study.

GBOL makes use of existing ontologies: PROV-O for activity capturing [291]; FOAF for agent information [81]; BIBO for article information stored within the annotation files [195]; SO for sequence information [154]; FALDO for genomic location [68], among many others. We refer the reader to [124] for detailed information on the integrated ontologies and the data model.

Annotations can be evaluated through critical examination of the provenance. The use of SPARQL allows complex queries across data annotated with SAPP and in direct comparison of these annotations with external resources, such as UniProt. Additionally for specific questions, likelihood values can be integrated, normalized or corrected for multiple testing. For instance, study of E-value distribution on instances of a protein domain across multiple genomes can inform optimal threshold selection, as shown in Fig.7.1C. SAPP implements existing tools: consistency of SAPP annotation and a comparison with deposited annotations is shown and discussed in [272].

By querying multiple consistently annotated genomes simultaneously, large scale functional comparisons can be performed without additional conversion steps (see Fig. 7.1D and [271]).

These examples demonstrate that by adopting FAIR principles to genome annotation, knowledge discovery is facilitated.

Funding

This work has received funding from the Research Council of Norway, No. 248792 (DigiSal) and from the European Union FP7 and H2020 under grant agreements No. 305340 (INFECT), No. 635536 (EmPowerPutida) and No. 634940 (MycoSynVac).

Chapter 8

The Empusa code generator: bridging the gap between the intended and the actual content of RDF resources.

Jesse C.J. van Dam, Jasper J. Koehorst, Peter J. Schaap, and Maria Suarez-Diez.

Abstract

The RDF data model facilitates merging of data that has been stored using different underlying schemas. This flexibility makes RDF an efficient alternative to develop resources integrating heterogeneous data sets. However, the lack of a predefined schema and the great flexibility of RDF might lead to a mismatch between the data structure in the resource and the ontology used to capture the data described in the resource documentation and schema. We have developed the Empusa code generator that from one Empusa definition format generates the ontology definition and API that can be used to validate the consistency of the exported data.

Introduction

Semantic Web technologies provide information retrieval and management systems to integrate heterogenous data from disparate sources [58]. The RDF data model is a W3C standard for storage of information in the form of self-descriptive subject, predicate and object triples that can be linked in an RDF-graph [79, 515]. The use of retrievable controlled vocabularies enables integration of data from different sources in a single graph and SPARQL can be used to query the so generated resources [22, 396]. RDF graphs have no predefined structure nor a schema, and the structure of an RDF resource can vary as new triples are added. Therefore, a formal definition of the relations among the terms, called an ontology, is required to efficiently retrieve linked information from these resources. Structural information can be encoded using Web Ontology Language (OWL) files [42]. RDFS is another related standard to define the structure of an RDF resource [80]. In this standard, each object can be defined as an instance of a class and each link as the realization of a property. Shape Expressions (ShEx) is a standard to describe, validate and transform RDF data. One of the goals of this standard is to create an easy to read language for the validation of instance data [69, 395, 468].

In a previous work, we developed RDF2Graph, a tool to automatically recover the structure of an RDF resource and to generate a visualization, ShEx file and/or an OWL ontology thereof [125]. Application of RDF2Graph to resources providing data in the RDF data model in the life sciences domain such as Reactome, ChEBI, UniProt, or those transformed by the Bio2RDF project [50, 55, 119, 219, 254] showed mismatches between the retrieved data structure and the one described in the OWL definition of each resource. The main reason for this lack of consistency is the great flexibility provided by RDF: as the data graph is free format, the ontology defines the structure but does not enforce it.

In the development of RDF resources, data export to RDF is often a source of errors such as typing errors in the predicates, instances with missing attributes, instances that did have a non-unique IRI, and instances that had no type defined, among others. Therefore tools that automatically store their output in RDF are essential to unlock the potential of these technologies. An example of such tools is the Semantic Annotation Platform with Provenance (SAPP) [273], that can automatically annotate genome sequences using standard tools and store the annotation results and their provenance in the RDF data model using the genome biology ontology language (GBOL) [124]. Development of such tools would be greatly facilitated by supporting tools able to read an ontology definition and generate code that can be used for data generation, export and validation.

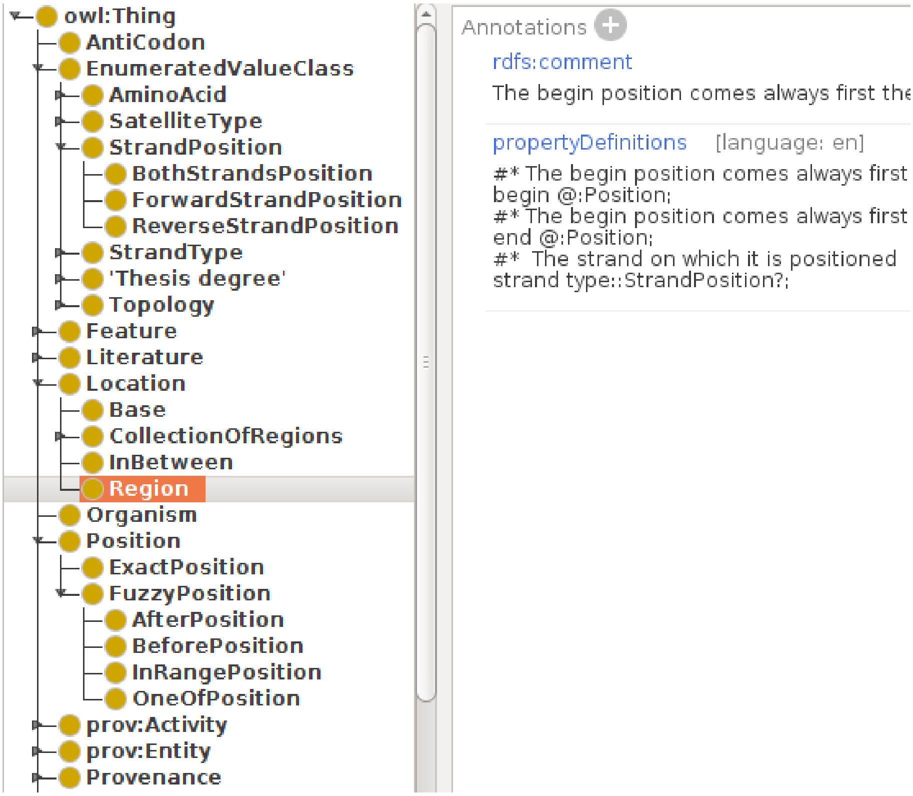
Here we present *Empusa*, that has been developed to facilitate the creation of RDF resources, which are validated upon creation. Empusa can be used to define an ontology and create an associated application programming interface (API) that can be used to perform data consistency checks. Therefore, the use of Empusa ensures consistency within and between the ontology (OWL), the Shape Expressions (ShEx) describing the graph structure and the content of the resource.

Implementation

The input definition of Empusa is a combination between OWL and a simplified version of ShEx, which can be edited within Protégé [347]. The classes are defined in OWL, whereas the properties are defined in each class under the annotation property 'propertyDefinitions' encoded within a simplified format of the ShEx standard. Additionally predefined value sets can be defined by adding a subclass to the EnumeratedValueClass. Each subclass of the value set is represented as one element within the value set.

The RDFS standard is used to define the *subClassOf* relationships between the classes, whereas the ShEx standard is used to define the properties of each class. Properties of the class are defined through the annotation property *propertyDefinitions*' as shown in figure 8.1. For each property the multiplicity and the expected type of the target value can be defined. The multiplicity can either be: *0..1* indicating that the property is optional and at most one reference is allowed; *1..1* indicating that one and only reference is allowed; *0..N* for optional properties with multiple allowed references; and *1..N* for properties that must have at least one reference. The '=' and '∼' sign can be used to define the references be stored as an ordered list to ensure that the elements are numbered. Target values types can also be defined. The type of the target value can be either: A simple value (String, Integer or Double, among others); Another class (for example a Protein); Or an IRI, referencing an external resource or ontology or to a sub-ontology(value set). Within the ontology, sub ontologies (value sets) can be defined under the EnumeratedValue class. Every sub class of EnumeratedValue class represents one sub ontology. All subsequent sub classes are elements of the sub ontology of which it is sub classed from. A class/sub class structure can be defined for these elements within the sub ontology.

The Empusa code generator uses this definition to generate: (i) An OWL file definition. It should be noted that the OWL file definition is generated as it remains general consensus within the field of semantics that these files are created for each ontology (ii) A full ShEx file that can be used to validate a data set containing information that is encoded with the ontology. (iii) An R and Java API, which one can use to generate the data with the encoding of the defined ontology. This API ensures that the multiplicities and referenced types are correct and prevents many errors in the data export. (iv) A canonicalized data format such that the data file encoded with the ontology can be read and modified by human editors. (v) A full documentation of the ontology based on *mkdocs*. The rdfs:label and skos:description properties can be used within the ontology to add a description about the classes and a comment line above each property definition in the simplified ShEx definition and can be used to add a description to each property.



rdfs:comment

The begin position comes always first the

propertyDefinitions [language: en]

```
#* The begin position comes always first
begin @:Position;
#* The begin position comes always first
end @:Position;
#*  The strand on which it is positioned
strand type::StrandPosition?;
```

FIGURE 8.1: **Example Empusa definition format.** Properties within a class can be defined with the propertyDefinitons annotation property. As an example, the the Region Class has been highlighted. Value sets (sub ontologies) can be defined under the EnumeratedValueClass class, for example the StrandPosition value set.

Results and Discussion

Empusa was developed primarily to help develop ontologies focussing on their function as a database schema for RDF resources. The design principles modularity, human readability, and annotation are followed to ensure that the so generated ontology can be easily extended [63]. Empusa can automatically and consistently generate an OWL and a ShEx definition, ontology documentation in markdown, an API, a JSON-LD framing file and a visualization. Empusa uses parts of the RDF2Graph tool [125] to generate a representation that can be subsequently used to generate a visualization within Cytoscape [444]. This allows users to browse the complete ontology intuitively.

Development of Empusa was closely related to the development of the GBOL stack [124] and the associated tool SAPP [273]. GBOL enables interoperable genome annotation, as it deploys and extends existing ontologies to represent genomic entities, their properties and associated provenance. The GBOL stack contains over 80.000 lines of R and Java code, OWL and ShEx definition files, and documentation files (mkdocs format). Generating such a large amount of code would entail 1 year of manual work (considering an efficiency of 50 lines per hour) [351]. The SBOL stack is a recently published platform for storing, publishing, and sharing synthetic biology designs [182]. GBOL contains 47673 lines of Java code for the API whereas the SBOL API has 27325 although it does also include some other supporting code. This shows the higher complexity of the GBOL ontology.

Moreover, during the development of the GBOL ontology countless updates were made to correctly encapsulate all the data and associated provenance. Most of these updates were based on insights gained through the data encoding process. Manually updating the code, without using the supporting Empusa tool, would have entailed so much work that it would still be an ongoing process. Thus, the Empusa code generator can serve to reduce the time (and costs) associated to development of ontologies and tools.

Conclusions

The Empusa code generator can be used to develop new ontologies combined with automatic generation of API and documentation. This reduces the complexity and time to extend and develop ontologies and tools able to exploit the full potential of Semantic technologies for heterogeneous data integration. Moreover, Empusa enables the validation of the generated resources and the verification of the consistency of the exported data thereby bridging the gap between the intended and the actual content of RDF resources.

Methods

Empusa is written in Java with Gradle as build system. Empusa is available at `http://www.gitlab.com/Empusa` under the MIT license. Documentation can be found at associated website `http://gbol.life/Empusa`.

Funding

This work has received funding from the Research Council of Norway, No. 248792 (DigiSal) and from the European Union FP7 and H2020 under grant agreements No. 305340 (INFECT), No. 635536 (EmPowerPutida) and No. 634940 (MycoSynVac).

Chapter 9

Comparison of 432 Pseudomonas strains through integration of genomic, functional, metabolic and expression data

Adapted from:

Jasper J. Koehorst*, **Jesse C.J. van Dam***, Ruben G.A. van Heck, Edoardo Saccenti, Vitor A.P. Martins dos Santos, Maria Suarez-Diez and Peter J. Schaap. "Comparison of 432 *Pseudomonas* strains through integration of genomic, functional, metabolic and expression data". In *Scientific Reports* volume 6, Article number: 38699 (2016)

*Equal contributions

Abstract

Pseudomonas is a highly versatile genus containing species that can be harmful to humans and plants while others are widely used for bioengineering and bioremediation.

We analysed 432 sequenced *Pseudomonas* strains by integrating results from a large scale functional comparison using protein domains with data from six metabolic models, nearly a thousand transcriptome measurements and four large scale transposon mutagenesis experiments.

Through heterogeneous data integration we linked gene essentiality, persistence and expression variability. The pan-genome of *Pseudomonas* is closed indicating a limited role of horizontal gene transfer in the evolutionary history of this genus. A large fraction of essential genes are highly persistent, still non essential genes represent a considerable fraction of the core-genome.

Our results emphasize the power of integrating large scale comparative functional genomics with heterogeneous data for exploring bacterial diversity and versatility.

Introduction

The *Pseudomonas* genus exhibits a broad spectrum of traits and *Pseudomonas* species show a remarkable adaptability to the biochemical nature of the large variety of environments, often extreme, they thrive in [498, 527]. The genus currently includes almost 200 recognized species, which have been clustered into seven groups and into lineages on the basis of a limited set of loci [304]. Some species are well-studied because they are human or plant pathogens, like *P. aeruginosa* or *P. syringae*, or because they are considered harmless and possess interesting biodegradation properties while others can produce a variety of extraordinary secondary metabolites with anti-microbial properties [207]. *P. putida* KT2440 is even Generally Recognized as Safe (GRAS-certified) for expression of heterologous genes and has been transformed into a genetically accessible laboratory and industrial workhorse [352].

A number of comparative genomics studies have been performed in the past [37, 304, 527] but the number of available *Pseudomonas* genomes quadrupled in the last five years due to the widespread use and the advancement of high-throughput sequencing technologies. As of December 2015, the complete and draft genomes of 432 strains distributed over 33 species are publicly available (see Supplementary Figure S1). This plethora of data entitles an in-depth comparative re-analysis of *Pseudomonas* genomes to explore their metabolic and ecological diversity.

Large scale functional comparison based on sequence similarity is challenged by methodological problems, such as the need of of defining arbitrarily generalized minimal alignment length and similarity cut-off for all sequence to be analyzed, and it is hampered by the high computational cost, since time and memory requirements scale quadratically with the number of genome sequences to be compared[271]. Many bacterial proteins consist of two or more domains and fusion/fission events are the major drivers of modular evolution of multi-domain bacterial proteins [382]. Interspecies domain variation can thus give rise to an annotation transfer problem: sequence based functional annotation methods use a consecutive alignment to identify common ancestry and therefore may miss domain insertion/deletion, exchange or repetition events, which may lead to functional shifts and promiscuity. Comparisons at protein sequence level should therefore be complemented with comparisons at the protein domain level [271]. In addition, in order to avoid technical biasses a biologically meaningful functional comparison requires consistent and up-to-date annotations. Instead, the biological information available in public databases varies in quality due to the use of different databases and annotation pipelines that include different methods and may assign different names, acronyms and aliases to the same protein. Re-interpretation of these predictions in most cases requires reverse engineering as data provenance is usually not available.

In this paper 432 Pseudomonas genome sequences were *de novo* re-annotated and the generated annotation information was integrated through a semantic platform with data from six metabolic models, nearly a thousand transcriptome measurements and four large scale transposon mutagenesis experiments. We identified phylogenetic relationships among different species using protein domains and performed extensive analysis of the core- and pan-genomes of the *Pseudomonas* genus and considered the habitat factor while analyzing the pan/core-genome. Finally, we linked domain content and domain variability of persistent and essential genes and their transcriptional regulation.

Results

De novo annotation of *P. putida* KT2440 as a minimal working example

P. putida KT2440 [352] is one of the best-characterized *Pseudomonas* strains. A *de novo* annotation obtained using an in-house annotation pipeline, the annotation deposited in GenBank (NC_002947) and an alternative annotation obtained using RAST [31] were compared, see Table 9.1. The total number of genes identified using three gene calling methods, Prodigal 2.6 (in our pipeline), Glimmer3 (RAST), and Glimmer (GenBank) are very similar, differing less than 4%. However, as each of these algorithms have an intrinsic false discovery rate in start-site prediction, significant differences in the start position of the identified genes were found. The number of exact matches in gene start-sites is only 73% (4073 genes) confirming previous observations [501]. These 5′ variations in gene identification can result in a putative gain or loss of biological functions; however, since different naming conventions are used in the different annotation protocols applied, a direct functional comparison to spot possible differences is not possible (Figure 9.1).

TABLE 9.1: Annotation results for *P. putida KT2440*. GenBank refers to the original deposited annotation (available at NCBI), whereas RAST and SAPP refer respectively to their annotation.

	#Genes	#Unique start/end positions	#Unique GO	Unique domains	Unique EC
GenBank	5350	170	0	3574	443
RAST	5531	62	726	3631	447
SAPP	5555	252	1403	3636	447

The use of controlled vocabularies overcomes this issue, so that functional comparison can be performed using gene ontology (GO) terms, Enzyme Commission (EC) numbers and InterPro identifiers. For the GenBank deposited annotation no GO information was available but the difference observed between the RAST and the *de novo* annotation is striking. This minimal working example shows that even for a single genome a comparative analysis of functional annotations derived from three work-flows is almost impossible

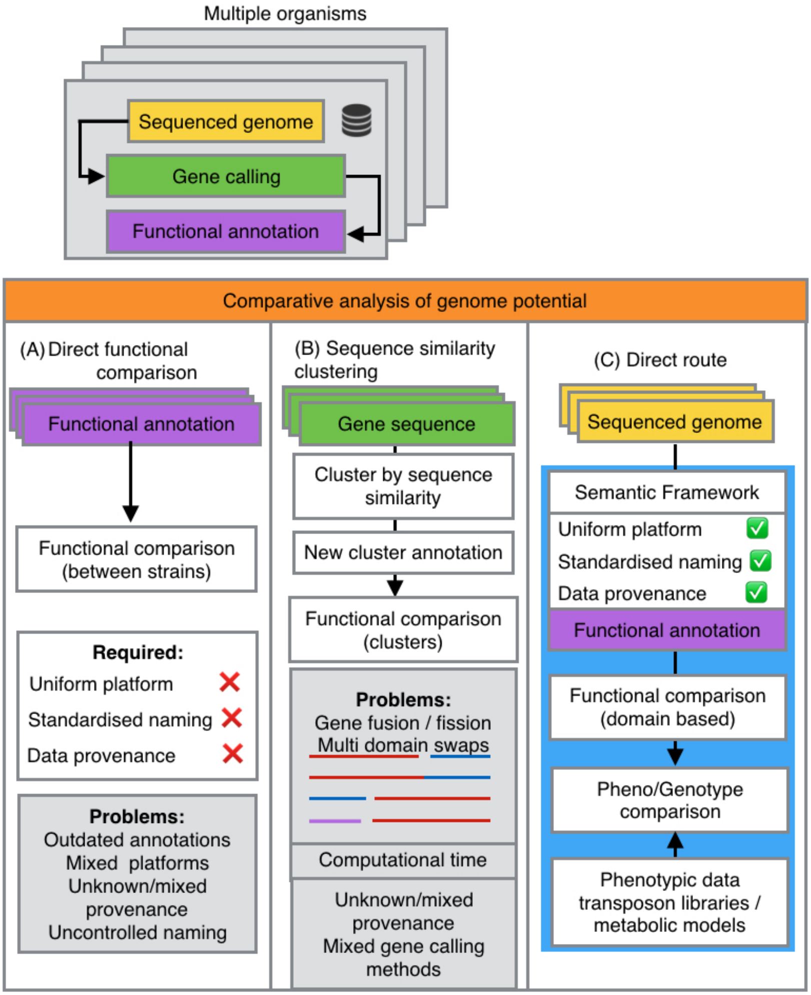


FIGURE 9.1: **Alternatives for functional genome comparison**: **A) Direct comparison of genome potential using existing annotation** is often hampered by lack of standardization of gene calling and annotation tools, mixed and unknown data provenance and inconsistent naming of function. **B) Sequence similarity clustering** bypasses inconsistent functional annotations. Computational time scales quadratically with the number of genome sequences and gene fusion/fission events might be overlooked. **C) Usage of standardised annotation tools** ensures uniform genome annotation prior to comparison; annotation provenance is stored for all steps.

by computational means due to lack of standardization and data provenance. This example further emphasizes that comparative genomic analysis requires homogeneous annotation.

Comparison of the genomic potential of *Pseudomonas* species

Since for a comparative genomics study a consistent and standardized genome annotation is a prerequisite, we evaluated the impact by comparing the functional annotations of 432 *Pseudomonas* genomes with a *de novo* annotation. We used both complete and draft genomes. According to the quality metric defined by Cook and Ussery, almost 30% of the available draft genomes were of low quality [116]. This was mostly due to a high number of contigs and not to the quality of the assemblies in itself, so they were included in the analysis.

GenBank files were converted into RDF, extracting genome sequences and gene-calls. Genomes were structurally and functionally re-annotated. The originally deposited gene-calls were functionally re-annotated as well and a pairwise comparison of GO terms, and EC identifiers assigned to the originally deposited and the *de novo* gene-calls was performed at gene and protein domain level. Figure 9.2 summarizes the results for the available 58 complete genomes. Differences in annotations were observed at all functional levels. Per genome on average 38 new genes were predicted while a functional re-annotation of the set of complete genomes yielded 838 additional GO-terms and 146 additional domains (For a more detailed overview see Supplementary Data S2). Considering the full set of 432 genomes, on average a difference of 153 genes per genome was detected. The results advocate for routine implementation of consistent gene-calling methods combined with an up-to-date functional annotation before performing comparative genomic analyses, as many of these differences will results in gain or loss of biological functions.

Sequence and function based comparative genomics of *Pseudomonas*

Genome-wide comparative analysis usually relies on sequence similarity clustering based on a blast-based all-against-all bidirectional best hit (BBH) heuristic approach. There are several limitations to this approach. Firstly, the runtime increases quadratically with the number and complexity of the species involved. Secondly, clustering is strongly context-dependent as it dramatically depends on chosen cut-off values to define statistical significance of sequence similarity. Problems may arise with in-paralogous sequences that evolve at very similar rates resulting from recent duplication events[357]. Thirdly, protein fusion and fission events are difficult to detect using alignments and thus critical information might be lost.

An alternative approach, already employed in a comparative genomics study of *Escherichia coli* [467], consists of grouping of proteins on the base of domain architectures with a fixed N-C terminal order []. Clustering based on

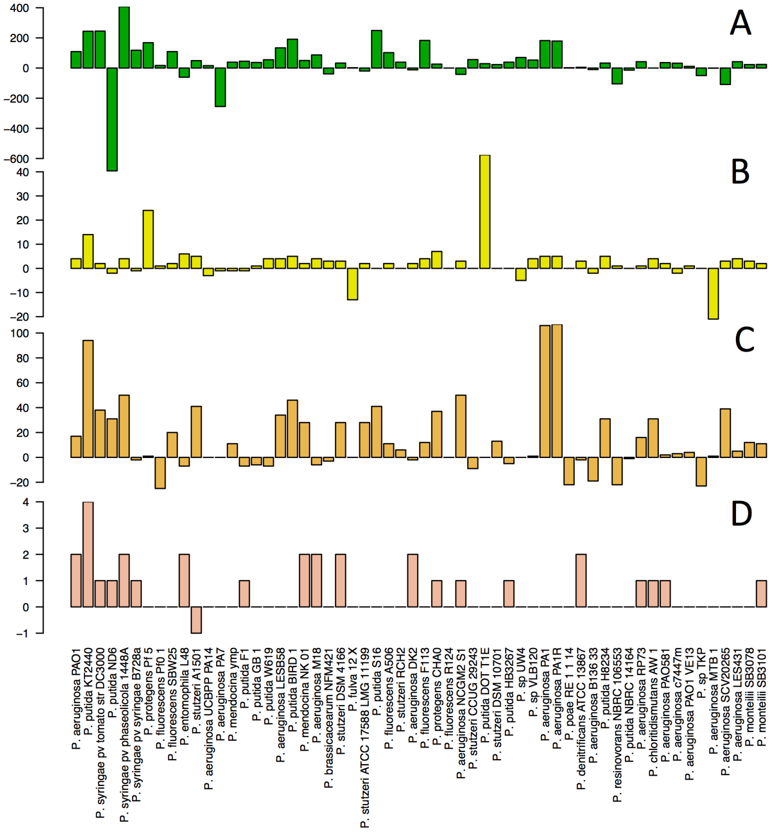


FIGURE 9.2: ***De novo* annotation of *Pseudomonas* genomes.** Comparison between the original and de novo annotations of 58 completely sequenced *Pseudomonas* genome sequences. Barplots indicate differences in the number of retrieved genome features terms between the *de novo* annotations and the original deposited annotations. A) gene abundance; B) protein domains; C: GO terms, and D: EC identifiers. The genomes are ordered from left to right by deposition date in the NCBI database (from oldest to newest).

domain order is highly scalable and moreover, most protein domains represent structural folds that can be directly linked to function. Here, both approaches were compared. Protein sequence similarity clusters were identified in a BBH approach using orthAgogue [155]. Due to runtime constraints, protein clustering was limited to the analysis of the 58 complete genomes leading to the identification of 14757 protein clusters. For each protein found within a cluster the domain content and N-C terminal domain order ranked by the position of the first detected amino acid of the domain (domain start) in the protein sequence (domain architectures) was analysed and is summarized in Figure 9.3A. 5515 sequence based protein clusters (37%) present a one-to-one correspondence to domain architectures, whereas 3134 (21%) can be associated to two distinct domain architectures. Overall, 93% of the identified clusters can be associated to 4 or less distinct domain architectures. Figure 9.3A also shows the number of proteins in each orthologous cluster. 3162 clusters (21%) contain proteins lacking established domains and almost 75% of them contain less than 10 sequences. These clusters correspond, in their vast majority, to hypothetical proteins. Regarding the core genome, 1618 clusters (11%) were found to be present in all 58 genomes. From these 1618 protein clusters, 242 contained duplication events leaving 1376 distinct single copy gene protein clusters common to all 58 genomes. 543 of those clusters showed a single domain architecture whereas the rest contained domain architecture variations as summarized in Figure 9.3C. We noted that such variability was mainly due to swapping or inversion in domains order. In a sequence based approach domain order variation can potentially lead to false negatives, broken clusters and even reduction of the core genome when more genomes are added to the analysis.

The analysis of 58 complete genome sequences showed that domain architectures retain enough information for functional characterization and that they can be used as a fingerprint for a functional cluster. Since the computational cost for obtaining protein domain identification scales linearly with number of genomes and can be easily distributed over multiple machines, we used these functional fingerprints to extend the analysis to all 432 *Pseudomonas* genomes. Over two million (2,704,339) genes were identified coding for over one million (1,196,884) unique protein sequences of which 85.6% (1,024,877) contain known protein domains. Figure 9.3B shows the results of persistence analysis, reporting the fraction of the total number of analysed genomes in which the corresponding cluster/protein domain/domain architecture was found; 40% the protein domains are persistent in the genus, showing that the functional information at domain level is preserved.

Classification of *Pseudomonas* strains based on genome potential

Patterns of protein domain presence/absence can provide an alternative and complementary way for assessing strain diversity [9, 531]. There are still many unclassified *Pseudomonas* strains and there is a continuous development on assessing the phylogeny using various approaches [60]. Figure 9.4 shows a distance tree of genome potential based on presence/absence of protein domains for the 58 complete *Pseudomonas* genomes. We found excellent agreement between this distance tree and the taxonomic classification based on 16S sequences indicating that binary patterns of protein domains retain enough information to reconstruct evolutionary history. The positioning of *Pseudomonas* sp. UW4 within the clade of *P. fluorescence*, confirms a previous observation

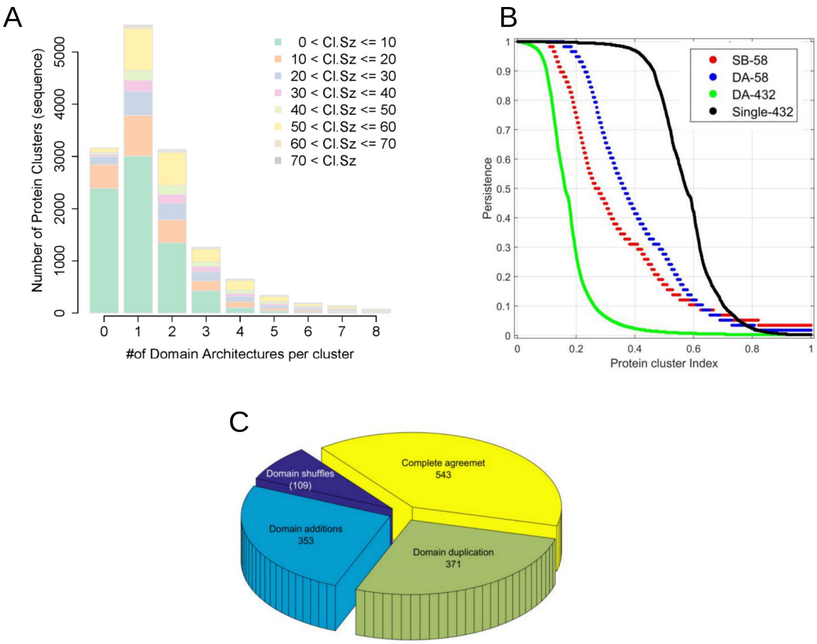


FIGURE 9.3: **Domain architectures in sequence based clusters of orthologous proteins** A) Number of distinct domain architectures per cluster B) Persistence analysis within the Pseudomonas genus. The curves indicate the persistence of each of the cluster. Clusters have been arranged by decreasing persistence values and the *x*-axis has been scaled to 0-1 range, in this way the cluster with the highest persistence have an *x* value of 0 and the cluster with the lowest persistence has an *x* value of 1. The *y*-axis indicates the persistence of a given cluster (see Equation 1): for instance a persistence of 0.8 indicates that 80% of the analyzed genomes contain sequences in that given cluster. SB-58 refers to the use of sequence based cluster considering the 58 complete genomes; DA-58 and DA-432 refers to the use of protein domains, for 58 and 432 genomes respectively; Single-432 reproduces the analysis for single domain proteins found in the full set 432 genome sequences. C)Variability in domain architectures per gene cluster in core-genome. Complete agreement indicates a unique domain architecture shared by all members of the cluster; For the cases where multiple domain architectures were found in a sequence cluster, the number of cases corresponding to domain duplications, additions and shuffles are indicated. (For A and B only 58 complete genome sequences considered).

based on 16S and three housekeeping genes (*gyrB*, *rpoB* and *rpoD*) [150]. *P. aeruginosa* and *P. stutzeri* clades are conserved while *P. putida* and *P. fluorescence* clades shows the addition of different species.

We further extended the domain based distance analysis to include all 432

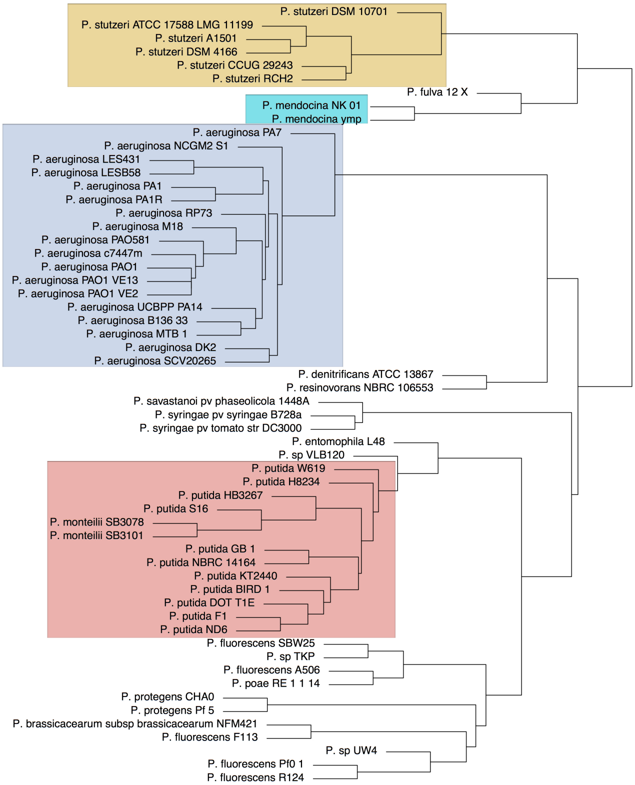


FIGURE 9.4: **Domain based distance tree of 58 *Pseudomonas* strains** The tree was build considering the pattern of presence/absence of protein domains using an average clustering approach. Only completely sequenced genomes are considered. The phylogenetic clusters corresponding to the most abundant species (*P. stutzeri, P. mendocina, P. aeruginosa and P.putida*) are colour-shadowed.

Pseudomonas strains (see Supplementary Figure S3). The majority of the strains cluster in accord with their taxonomic classification. Many of the unclassified strains could be classified either in *P.aeruginosa* (4) or *P. putida* (13).

Exploring the pan- and core-genome of *Pseudomonas* at protein domain level

The core-genome of a taxon level is defined as the genes persistently present in the population, while the pan-genome is essentially the amount of different genes found within a population at the specified taxonomic level [465]. The currently available genomes allow to measure the pan- and core-genome sizes, however these sizes change upon the addition of new sequences. The core-genome is usually reduced and the pan-genome increases mostly due to the discovery of novel accessory genes that accumulate by lateral transfer, forming new trait combinations until saturation has been reached. Saturated pan-genomes with a stable core-genome are called closed. From the currently available genomes an estimation can be made, using mathematical modelling [465], of the size of the pan- and core- genomes that are expected if the sequences of every existing strain were to be included in the analysis. We refer to these estimations as estimated pan- and core- genome sizes.

Genome potential of the genus *Pseudomonas* is reflected in its metabolic diversity which allows individual species to inhabit a wide variety of environments. With the current set of 432 (draft) genomes we studied whether the observed diversity in genome potential reflects a closed pan-genome. We initially considered the 58 complete genomes. Observed core-genome of 2687 protein domains was to be confronted with an estimated size of 2681. For the pan-genome we found 6472 protein domains (observed) versus 6541 (estimated). Since these measures depend on the number of genomes considered, we explored how these measures vary by using a different number of genomes (from 5 to 58). This was achieved by applying a 10-fold random re-sampling from the 58 genomes to obtain an indication of the possible variability (Figure 9.5). As expected the size of the core-genome of the genus decreases with the number of genomes considered while that of the pan-genome increases. The observed and estimated sizes of both the pan- and core-genome are rather stable with respect to the number of genomes used in the calculation, except for small sample size (< 15).

Including draft genomes in the calculations resulted in a dramatic reduction, up to the 73%, of the size of the core-genome both observed and estimated, which dropped to 726 and 720 protein domains architectures, respectively. Interestingly, this reduction does not lead to a loss of functional information since single domains are highly persistent as previously stated (40%).

We observed a large variability for both measures. The reduction of the core size and its variability can be partly explained due to the inclusion of draft genomes with a high number of gaps containing non-sequenced genes. The difference between observed and estimated sizes reduced to only one protein domain for both the pan- and core-genome, indicating saturation. Addition of new genome sequences to the analysis will most likely not lead to the identification of a significant set of new domains within this genus. This saturation effect does not depend on the particular estimation model used. Saturation of the pan-genome was also seen through a heap model ($\alpha = 1.30 \pm 0.05$). In this analysis values > 1 indicate a closed pan-genome [491].

Essentiality analysis of domains in the core-genome

From a functional point of view, the core-genome of a genus is most likely enriched in essential genes necessary for (long term) viability and adaptation to ever changing environmental conditions. Since persistence can be used to

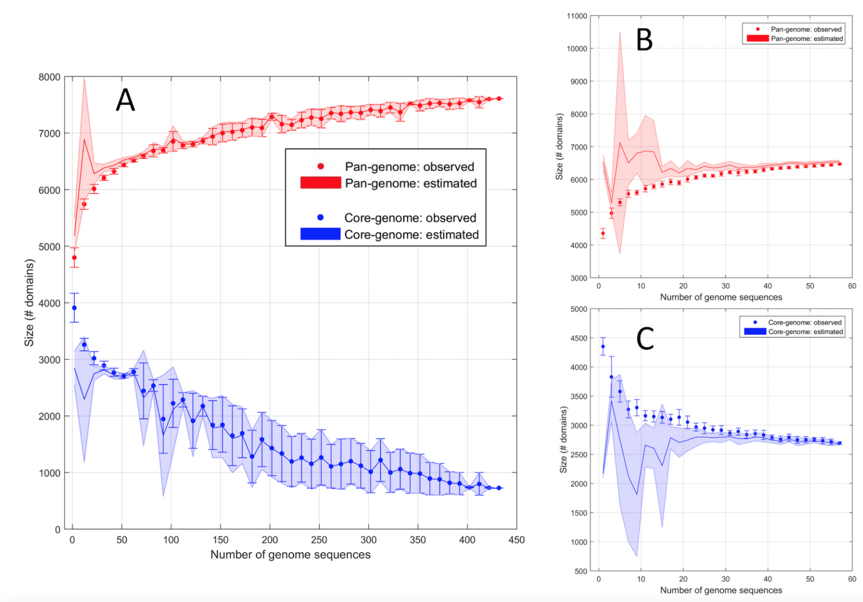


FIGURE 9.5: ***Pseudomonas* pan- and core-genome defined on the base of protein domains** A) Complete overview of the distribution of the size of the pan- and core-distribution of protein domains. Error bars correspond to standard deviations based on 10 measured random realizations of the indicated number of genomes whereas the shadowed area is the estimated standard deviation using the same approach. B) Pan-genome of the 58 fully circular genomes. C) Core-genome of the 58 fully circular genomes.

identify genes required for survival [3, 329], a positive correlation between persistence (the number of genomes sharing a given gene) and essentiality can be hypothesized. To verify this hypothesis we combined gene essentiality measures with gene persistence in the genus. Gene essentiality was defined from experimental results available for two *P. aeruginosa* strains (PAO1 and PA14)[294, 300] and from *in silico* predictions. For the latter, we considered 6 genome-scale constraint-based metabolic models which rely on functional annotation to uncover the metabolic potential of biological systems and are able to accurately predict gene essentiality in a large variety of growth conditions [367].

We observed that essential genes show higher persistence values than non essential ones: this relationship is conserved when persistence is computed either using a sequence similarity based approach on 58 completely sequenced genomes or for 432 genomes by using a domain architecture approach as shown in Figure 9.6A.

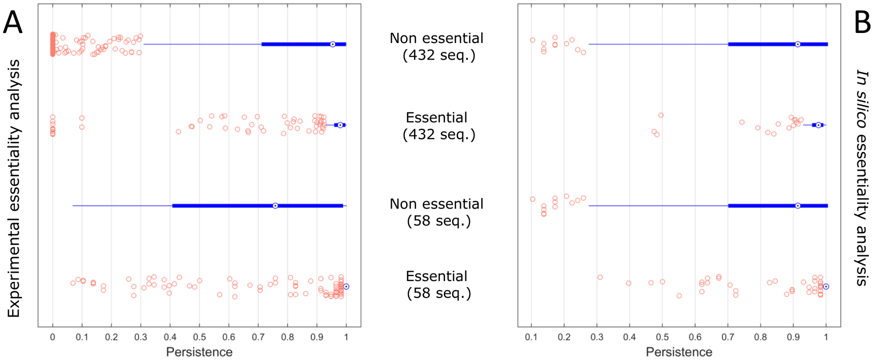


FIGURE 9.6: **Persistence of (non) essential genes**. A) Persistence of essential and non-essential genes as derived by experimental investigations. B) Persistence of essential and non-essential genes as derived by *in silico* modelling using genome based constrained metabolic modelling. Results shown pertain the use of the iMO1086 model for *P. aeruginosa* PAO1. In both cases persistence is calculated using the 58 completely sequenced *Pseudomonas* genomes and the complete set of 432 genomes sequences. Magenta (circle) dots indicate outliers.

A comparison of gene persistence and essentiality for the two strains showed that 65% of genes found to be essential for PA14 growth on LB are also essential for growth of PAO1 on either LB, minimal with pyruvate or sputum agar, but only 39% of genes reported to be essential for PAO1 growth were found to be essential for PA14 (See Supplementary Figure S4). This difference could be due to the smaller set of tested conditions. We used a less stringent cut-off for persistence: 0.95 instead of 1 to allow for non-sequenced genes due to incomplete draft genomes. Therefore, we observed that a small fraction of persistent genes is present in only one of the two strains (0.016% and 0.025% for PA14 and PAO1, corresponding to 75 and 47 genes respectively) which are likely to have been lost through evolution.

Analysis of the complete pan-genome revealed that 1252 single copy genes are persistent. Of these, almost one third (404) were found to be essential *in vivo* under three growth conditions (LB, minimal-pyruvate or sputum agar) for *P. aeruginosa* PAO1 strain []. Similar ratios were observed for strain PA14.

1112 unique domains were identified in the 404 essential persistent genes and 1340 unique domains in the non-essential but persistent genes. 203 domains were shared between essential and non-essential persistent genes. Essential genes contain a larger repertoire of unique, single copy domains: 404 essential persistent genes contained, on average, 1.53 single copy domains whereas for non essential persistent genes, the average was 0.82.

In vivo essentiallity analysis were limited to four conditions. Using metabolic models a wider range of conditions can be explored albeit the analysis is restricted to metabolic genes. We considered six genome scale constraint based metabolic models describing the metabolism of *P. aeruginosa* PAO1 (models iMO1056[359] and iMO1086 [360]), *P. fluorescens* SBW25 (iSB1139 [71]) and *P. putida* KT2440 (iJN746 [356], iJP815 [397], and iJP962 [360]).

We explored a wide range of growth conditions with varying carbon, nitrogen, phosphorus and sulphur sources and for each medium composition, gene essentiality predictions were performed using Flux Balance Analysis and are summarized in Table 9.2. Figure 9.6B shows results for *P. aeruginosa* model iMO1086, confirming what was observed for experimental data. Of the 750 essential metabolic genes that were identified under 3366 media compositions for iMO1086, 169 genes were identified to be essential under experimental conditions whereas 42 genes were essential but not *in silico* (25%). Average persistence over the 58 complete genomes was 0.96±0.14 for predicted essential genes and 0.85±0.24 for non-essential, which we found to be significant (p-value < 0.01 for a Wilcoxon test). When considering the 432 genomes, we still observed difference in the persistence of predicted essential and non essential genes 0.95±0.12 versus 0.89±0.21, p-value < 0.01). Similar results were also obtained when using essentiality predictions for the other metabolic models.

TABLE 9.2: Conditional gene essentiality predictions using six metabolic models from three *Pseudomonas* species.

Organism	P. aeruginosa		P. putida			P. fluorescens
Model	iMO1056	iMO1086	iJN746	iJP815	iJP962	iSB1139
Medium sources						
#Carbon	49	51	60	40	43	44
#Nitrogen	32	33	22	25	27	19
#Sulfur	4	1	10	1	1	6
#Phosphor	2	2	1	1	1	2
Genes						
#Essential/ persistent*	115/106	149/132	118/104	112/100	162/148	117/95
#Conditional/ persistent*	591/278	601/278	389/170	113/64	495/252	615/290
#Non-essential	348	336	253	593	305	407
#Overlapping genes*	95		68			

*Persistence was computed for each essential and conditional essential genes over the 58 *Pseudomonas* genomes

Using metabolic models to simulate media compositions we identified additional genes that were essential in a number of conditions, retrieving on average 1.47 single copy domains per gene, consistently with what observed for essentiality experiments. We further combined the models' predictions and we inspected genes predicted to be essential in all the tested conditions. For *P. putida*, the three models showed an overlap of 68 essential genes. Interestingly, these genes contained 2.53 single copy domains on average, underpinning previous results. Non-essential genes contain domains that are shared with other genes. This can result in the presence of isozymes or of potentially moonlighting enzymes which can step in for essential functions in the case of deletions or mutations.

Variability of gene expression and its association to persistence and essentiality in *Pseudomonas*

Associations between gene essentiality and low variation in protein abundance have been observed in *E. coli* [487]. We hypothesized the existence of an association between gene persistence and expression level variation. We analysed gene expression variability in *P.aeruginosa* using a gene expression compendium containing over 900 samples and 100 datasets regarding *P.aeruginosa* PAO1 genes [485]. Each gene was assigned a score, Variability, for transcriptional variation. Persistent genes tend to show significantly lower degree of variation in expression level than non persistent ones (p-value < 0.01); this holds true also for essential genes (Figure 9.7). Similar results are obtained when analysing a more limited dataset containing RNAseq measurements of *P.aeruginosa* PA14 in 14 growth conditions[149] (see Supplementary Methods S5) This association between low expression variability and persistence/essentiality could indicate that expression of genes in the core-genome is likely to be buffered and independent from environmental growth conditions. To the best of our knowledge such associations have never been established on such large scale due to the limitations associated to comparing hundreds of genome sequences.

Discussion

For our analysis we did not rely on previously existing annotations, but we performed a consistent re-annotation of all the sequences using a standardized approach that ensured coherence and uniformity. A sequenced based approach was used for a prior comparative analysis to define clusters of orthologous proteins in the smaller dataset of 58 complete genomes. Due to polynomial growth of computational time, this approach is not feasible for large data sets. Mining a gene sequence for domain occurrences is less computationally demanding, which provides an effective scalable approach.

Sequence based approaches are used to identify clusters of orthologous proteins, however the analysis of domain architectures is targeted towards the identification of groups of functionally equivalent proteins. Protein domains provide a standardised way to assess sequence variation and its impact in function, since every amino acid has a characteristic weight in the domain model. Protein domains are more strongly associated to protein structure than protein sequences, thereby providing a closer link to function that can bridge over larger evolutionary distances, which is essential to comparative functional analysis. Still there is a need for improving how protein domain are defined to accommodate similar models arising from, possibly different,

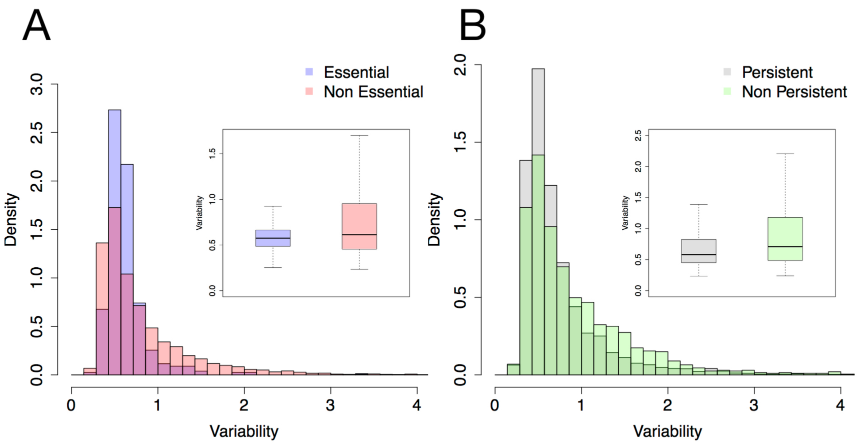


FIGURE 9.7: **Variability of gene expression levels and its association with persistence and essentiality** A) Distribution of Variability score for (non) persistent genes (genes with persistence lower or higher than 0.95, respectively). Box plots show Variability values for both groups. Difference between mean values is significant (p-val < 0.01). B) Distribution of Variability score for essential and non-essential genes with gene essentiality derived experimentally [293]. Box plots show Variability values for both groups. Difference between mean values is significant (p-val < 0.01).

databases and to take into account positional variations that might lead to spurious domain inversions.

When applied to the inferred proteomes of the 58 complete genomes, both clustering methods yield similar results. The same clusters were obtained in 40% of the cases meaning that each of these clusters contained an equal number of proteins, captured the same strains and shared the same domain architectures. In 20% of the cases, very similar but numerically distinct clusters were obtained, as a given sequence similarity cluster had captured two distinct domain architectures. In most of these cases variability in domain architecture were caused by changes in domain order due to small variations in the start position of overlapping domains. Approximately, 20% of identified proteins have no recognizable functional domains. As most of these proteins are hypothetical they were not considered for functional analysis. When only proteins containing domains are considered, over 90% of the clusters identified using sequence comparisons contain 4 or less distinct architectures.

The differences in the persistence curves shown in figure 9.3C show that the way the clusters are defined, either using sequence similarity or protein domains, impacts the calculation of gene persistence: this has repercussions on the definition of the core genome and its size. We found these differences to be larger when more genomes are considered. This is more likely linked to the broader range of phylogenetic distances among considered genomes: this is explored in more detail in Koehorst *et al.*[271]

Our analysis resulted in the identification of the pan- and core-domainome of 432 *Pseudomonas* which is closed according to the heap model as also recently noted for the *P. aeruginosa* species [345]. This suggests that sequencing additional strains will fail to add new genes to the pan-genome: however, this is likely an oversimplification. Here, we understand closeness of the pan-genome as measure of the genus ability to acquire exogenous genes and as a proxy for the ratio between vertical and horizontal gene transfer indicating that horizontal gene transfer has not played a major role in shaping the genome content of the genus.

Key characteristics of *Pseudomonas* must be located in the genus core-genome, however comparison with metabolic models shows that identified core is not autonomously functional. Not all the genes in the core-genome seem to be essential (under given tested conditions), however essential genes represent $\approx 40\%$ of the core-genome, in agreement with previously reported ratios for other species/genus [532]. The remaining 60% contain unique features defining the genus.

We found a strong association between gene essentiality and protein domain properties. We observe an inverse correlation between the number of proteins in the genome containing the considered domain and essentiality, with average number of domains uniquely present in the considered protein going from 1.5 to 0.8 when non essential/essential genes in the core-genome are considered. The average number of single copy domains per gene further increases when stricter criteria for gene essentiality are applied, namely that genes should be essential in all the simulated media.

Accurate algorithms to predict gene essentiality from genomic features have been also developed and domain enrichment score has been shown to have a high predictive power[134] which is computed based on the ratio of occurring frequencies of a particular domain between essential genes and the total genes in the whole genome of already characterized species. Here we have established a link between the number of copies of a domain in a genome and gene essentiality that can be used to complement essentiality predictions.

The extensive use of metabolic reconstructions allowed us to identify conditionally essential genes, and a large number of single copy domains is also observed in these genes. This supports the idea that protein domains are the driving force behind gene essentiality which is preserved through protein domains rather than through the conservation of entire genes [135].

We have shown that lower fluctuations in gene expression are associated to essential and/or persistent genes. Further work is required to clarify the overlap and intertwining between both gene categories (essential/persistent) and to clarify the (possibly different) regulatory mechanisms stabilizing their expression levels.

Methods

Genome retrieval

Genbank files containing genome sequences and existing annotations for 58 circular genomes and 374 draft genomes of the *Pseudomonas* genus were downloaded from the GenBank database in June 2015. Annotation of *Pseudomonas* KT2440 was also downloaded from RAST [31]. A detailed list of the included strains is available (see Supplementary Figure S1 and Supplementary Data S2).

Genome de novo annotation

To perform the re-analysis of the 432 genomes sequences we used a in-house pipeline for annotation and data storage[271]. Likewise existing annotation pipelines such Prokka [437], it relies on external feature prediction tools to identify the coordinates of genomic features within genomics sequences. The pipeline consists of a number of python modules that execute annotation applications and convert results and provenance directly into the RDF data model with a self defined ontology (the complete description of the implemented ontology can be obtained using RDF2Graph [125]) using the RDFLib library. For genetic elements determination a variety of tools is implemented such as Prodigal [235] for gene prediction. The main difference is that results are stored as Turtle files [52] containing an RDF model which allows simultaneous exploration of annotation data of multiple genome sequences, greatly facilitation multiple comparison and the integration of heterogeneous source of information. Since it deploys semantic features allowing the storage of data provenance, we refer to it as SAPP (semantic annotation pipeline with provenance). Annotation can be exported to other formats for downstream processing with other tools such as Roary [372]

Each genome sequence was converted to the RDF data model using the EMBL/GBK to RDF module. *De novo*. The FASTA2RDF, GeneCaller (a semantic wrapper for Prodigal 2.6 [235]) and InterPro (a wrapper for InterProScan [249]) modules were used to handle and annotate the genome sequences. Results were retrieved with SPARQL queries.

Protein domain presence and phylogenetic analysis

A SPARQL query was used to extract the presence of protein domains for all 432 genomes. Data were stored in a 432 (genomes) by 7608 (protein domains) binary matrix (0/1 for absence/presence). Protein domains were identified by their InterPro identifiers. Phylogenetic trees based on protein domains were created taking as input the domain presence/absence matrix. The R package `pvclust` was implemented in R (version 3.3.1)[399] with a binary distance and average clustering approach with a bootstrap value of 10 [482].

Protein domain architecture based clustering

The positions (start and end on the protein sequence) of domains having InterPro [248] identifiers were used to extract domain architectures (*i.e.* combinations of protein domains). Protein domains were retrieved for each protein individually. The domain starting positions were used to assess relative position in the case of overlapping domains; alphabetic ordering was used in the case of domains with the same starting position. Labels indicating N-C terminal order of identified domains were assigned to each protein so that the same labels were assigned to proteins sharing the same domain architecture. Here we have followed a strict approach and two domain architectures were considered different whenever they had different domains or they appeared in different order. For more details see Koehorst *et al.*[271].

Estimation of pan- and core-genome size

The estimated number of domains in the pan- and core-genomes expected if the sequences of every existing strain were to be included in the analysis were computed using binomial mixture models as implemented in the `micropan` R package [466] using the domain presence/absence matrix previously defined and default values for the parameters. *Pan-* and *core-* analysis was initially performed on the 87 genomes with a maximum of 3 contigs to avoid bias due to incomplete genome sequences. Analysis was extended to the remaining 374 draft genome sequences available. To obtain an indication of the variability of these measures as function of the number of sequences used, these were calculated by a 10 fold random sampling from the full set. Heap analysis as implemented in the `micropan` R package was used to estimate openness or closeness of the pan-genome [491] using 500 genome permutations and repeating the calculation 10 times. Final measure is given as the mean ± standard error.

Orthologous gene detection

Orthologous genes were calculated initially for the set of 58 completely sequenced genomes. Protein sequences predicted using Prodigal 2.6 were extracted using a SPARQL query and used in a Best Bidirectional Hit approach [489]: using an all-versus-all BLASTP comparison and an E-value threshold of 10^{-5} and a maximum target sequence of 10^5. OrthAgogue [155] was used to convert BLAST results into a weighted graph. The MCL [159] clustering algorithm was applied, using an inflation value of 1.5, on the graph to define protein clusters. The results were then extrapolated to the full set of 432 genomes using cluster specific domain fingerprints. Specifically, the sequence clusters obtained through MCL clustering on the 58 complete genomes were used to define sets of protein domains (each sequence cluster was mapped to a set of domains). The remaining genomes were then looked for any given domain set defined on the 58 genomes to define their presence/absence in the draft genomes.

Persistence and essentiality analysis

The persistence of a gene can be defined as

$$Persistence = \frac{N(\mathrm{orth})}{N} \tag{9.1}$$

where $N(orth)$ is the number of genomes carrying a given orthologue and N is the number of genomes searched [162]. For the 58 completely sequenced genomes, orthologous genes were inferred using a BBH approach. For the full set of 432 sequenced genomes orthologous genes were inferred by making use of protein domain arrangements.

Locus tags for predicted proteins were inferred from the original annotation through SPARQL. Locus tags were linked to gene essentiality as defined in experimental studies available for *P. aeruginosa* PAO1 [] and PA14 [300]. For each of the predicted proteins with inferred locus tag the corresponding protein cluster was initially calculated for the 58 genomes. The domain architecture corresponding to each cluster was extracted and subsequently scanned against all 432 available sequences. We used the MCL clusters as a reference set for the identification of domain architecture variations which were then extrapolated over the 432 genomes. The persistence for each locus tag was calculated and compared against the essentiality score obtained from two experimental studies.

Metabolic model essentiality analysis

We considered six genome scale constraint based metabolic models describing the metabolism of *P. putida* KT2440 (models iJN746 [356], iJP815 [397], and iJP962 [360]), *P. aeruginosa* PAO1 (models iMO1056[359] and iMO1086 [360]) and *P. fluorescens* SBW25 (model iSB1139 [71]). For each genome-scale metabolic model we performed a single gene essentiality analysis in a large number of growth media varying in carbon (C), nitrogen (N), phosphorus (P) and sulphur (S) source. To define the growth media we first identified candidate C, N, P, and S sources in each model independently. Because chemical sum formulas were not always available, we considered each compound for which an exchange reaction was present as a candidate C, N, P and S sources. We changed the *in silico* medium composition to a minimal salts medium containing glucose as C source, ammonia as N source, phosphate as P source, sulphate as S source, in addition to oxygen, water, H$^+$, and a variety of salts depending on the particular model considered. The potential of each candidate C, N, P, and S source was then evaluated by adding it to the *in silico* medium while omitting the default C, N, P, or S sources. Growth predictions were performed using Flux Balance Analysis [367] as implemented in the Matlab COBRA Toolbox [434]. This provided 4 lists of compounds that were suitable as C, N, P or S sources which were then combined into a single list of growth media by taking all combinations of compounds from the 4 lists. For each medium, we then used the *singleGeneDeletion* function from the COBRA toolbox to determine the growth rate of the mutant strains. If a gene knock-out reduced the *in silico* growth rate below 10^{-6} we considered the gene as essential. Models and Matlab scripts used in this analysis are available in Supplementary Data S6.

Comparison of gene expression profiles

A publicly available gene expression compendium for *P. aeruginosa* was retrieved [485]. Briefly, this dataset contains a collection of gene expression datasets (950 individual samples pertaining 109 distinct datasets) measured using Affymetrix platform GPL84 and processed using a common normalization and background correction protocol. The final dataset contains expression measurements (in a log_2 scale) for 5549 genes from *P. aeruginosa* PAO1. For every gene we considered its expression profile in this compendium and a Variability value was calculated as the ratio between the standard deviation and the mean.

Availability of Data and Materials

The annotation pipeline framework is distributed under the MIT license. The pipeline all genomic data, data provenance and computational results associated with this study are freely available at `http://semantics.systemsbiology.nl`. Additionally, the data associated to this study are provided in turtle format as an RDF serialized dump. This dataset is made available under the Open Database License: `http://opendatacommons.org/licenses/odbl/1.0/`.

Acknowledgements

This work was supported by the European Commission-funded FP7 project INFECT (contract number: 305340). This work was carried out on the Dutch national e-infrastructure with the support of SURF Foundation.

Author contributions

J.J.K, J.v.D, V.M.d.S and P.J.S participated in the conception and design of the study. J.J.K and J.v.D were responsible for the code and design of the semantic framework. R.v.H performed model-based essentiality analysis. M.S-D performed the integration of expression data. J.J.K, E.S, V.M.d.S, M.S-D, and P.J.S wrote the manuscript. All authors critically revised the manuscript.

Additional information and files

All additional files can be found at the on-line version of Jasper J. Koehorst*, **Jesse C.J. van Dam***, Ruben G.A. van Heck, Edoardo Saccenti, Vitor A.P. Martins dos Santos, Maria Suarez-Diez and Peter J. Schaap. "Comparison of 432 *Pseudomonas* strains through integration of genomic, functional, metabolic and expression data". In *Scientific Reports* volume 6, Article number: 38699 (2016)

Chapter 10

Bio-Growmatch: high quality automatic model building through automated incorporation of phenotype data

Jesse C.J. van Dam, Vitor A.P. Martins dos Santos, Peter J. Schaap and Maria Suarez-Diez

Abstract

Genome-scale metabolic models have been proven essential to unravel bacterial metabolism and to predict metabolic phenotypes from genome information. The successful applications of these models to metabolic engineering has boosted the development of tools for automated model reconstruction. However, still building a genome scale model is a labour intensive process entailing manual revision of hundreds of reactions. Gap-filling algorithms have improved the quality of the automated reconstructions thereby reducing the amount of curation needed. Still often the improved models are not able to fully account for known metabolic phenotypes.

Here, we present Bio-Growmatch, a gap-filling tool that incorporates available phenotype information in the reconstructed draft, thereby largely improving their quality. We have tested the performance of Bio-Growmatch on 31 bacterial species. For these we have performed high-troughput measurements using phentoype microarrays. The obtained data have been used to evaluate the improvements introduced using Bio-Growmatch on draft models generated using two popular tools for automated model generation (SEED and PathoLogic) and different requirements for model completion. Overall, we have tested the performance of Bio-Growmatch in more than 3000 models and found that it improves the predictive power of the models.

Introduction

High resolution sequencing of bacterial genomes has become very affordable, single cell genomics is becoming commonplace and the quality of metagenomics datasets has increased considerably. As a result, the number of published bacterial genomes has grown exponentially [112, 474]. This results in an increased demand for computational tools to predict bacterial phenotypes from the genotype. In this respect, genome-scale metabolic models (GEM), based on stoichiometric constraints, can predict metabolic phenotypes such as catabolic potential, essential media components, alternative culture conditions and conditional gene essentially [24, 87, 272, 537].

GEM condense the full set of biochemical reactions that can occur in an organism. The identified reactions and their corresponding metabolites are stored in a matrix formalism suitable for mathematical analysis [358, 494]. The starting point of a reconstruction is a comprehensive list of genome encoded enzymes and associated reactions and metabolites. Building a GEM is a labor intensive process. Currently, a number of software tools exist that automatically mine reference reaction databases to produce a draft of the desired model [216].

High quality automatically generated drafts greatly diminish the amount of labor for GEM creation. Such draft models are often incomplete and contain errors and inconsistencies but provide a solid starting point for subsequent manual curation and improvement. Often, these inconsistencies appear as gaps in the model that lead to blocked reactions, that are not able to carry any flux in any condition or as dead-end metabolites that can only be either produced or consumed. These gaps prevent the use of many techniques, such as Flux Balance Analysis (FBA) [367], commonly used to analyse GEM. FBA returns a prediction on the optimal (maximum or minimum) flux value through a selected reaction, called objective function. Growth predictions are commonly performed by selecting biomass formation as objective function. However, FBA is only possible when all the reactions connected to the selected objective function are present in the model.

Gap-filling has been defined as "a computational technique to complete a reaction network based on FBA without referring to the genome" [289]. Computational tools to perform gap-filling, such as Gapfill [432], Smiley[407] or fastGapFilling [289] use no other input than a draft model and a set of constraints defining possible media. Other tools incorporate additional experimental data: C13 labeling data, expression datasets, knock-out data or growth data on various substrates [47, 223, 225, 432, 544, 545]. Phenotype data characterizing growth and substrate utilization are relatively easy to obtain and can be obtained in a high throughput manner [67]. However, to the best of our knowledge, no workable implementations are available for any of the tools that use this data type.

We have developed Bio-Growmatch, a modification of GrowMatch that is specialized on using metabolic phenotype data to gap-fill and improve an automatically generated GEM. One of the major problems encountered when gap-filling a GEM pertains how biomass is described. The need of gap-filling techniques often arises as a result of missing anabolic reactions needed to synthesise biomass components. On the other hand, substrate utilization predictions rely on completeness of catabolic pathways. Thus, the prediction of growth and substrate utilization on a given medium is dependent on both the catabolic and anabolic capacities of an organism. Within Bio-Growmatch, it is assumed that these are to some extent independent from each other. Two main approaches are followed i) a set of relatively simple biomass definitions is used, which limits the requirements on anabolic capacities and increases the likelihood of a positive growth prediction. ii) Bio-Growmatch performs a gap-filling using tricarboxylic acid (TCA) cycle intermediates as media, which we named as the anabolic fast gap fill method. This effectively decouples the gap-filling of anabolic and catabolic pathways.

In addition to a working implementation of Bio-Growmatch, an extensive evaluation of the performance of the algorithm is presented. 31 bacterial species, with available genome sequences, have been selected. For these, draft network reconstructions have been generated using combinations of two approaches for genome annotation (SAPP and RAST [31, 273]), two automated draft generation tools, namely SEED and PathoLogic [264, 371], which use two reference reaction databases: ModelSEED and MetaCyc [95, 137]. High-throughput data on the metabolic phenotypes of these organisms has been generated using Biolog phenotype microarrays [446]. Subsequently, these experimental data have been compared with the predictions produced by the models, obtained before and after using Bio-Growmatch. A scheme of the tested combinations is presented in Figure 10.1. Furthermore, we have also tested the improvements introduced by Bio-Growmatch in four published and manually curated models. Overall, we have tested the performance of models generated by combination of two model generating methods and their associated gap filling methods, four biomass definitions, the inclusion of the anabolic fast gap-filling method and our new Bio-Growmatch method.

Results

Bio-Growmatch

Bio-Growmatch is based on the same principles as the algorithm presented in [223], which in turn relies on the principles demonstrated by GrowMatch [281]. Bio-Growmatch has four main steps, as shown in Figure 10.2. In the first step (gap-filling) reactions are identified that, upon addition to the model, would reduce the number of false negative (FN) predictions (no growth or no degradation of compounds predicted when growth or degradation is experimentally proven to occur). This produces a set of candidate reactions that, in the second step, are reconciled to minimize the number of reactions added to the model. In the third step (gap-creation) reactions are identified that, upon removal from the model, create gaps that reduce the number of false positive

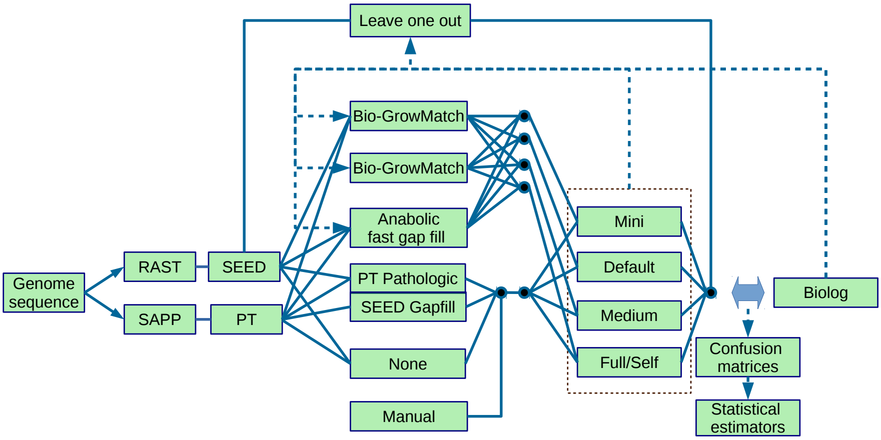


FIGURE 10.1: **Tested combinations.** Schematic overview of all the tested combinations: Two genome annotation pipelines (SAPP ([273]) and RAST([31])), two automatic model generation tools (SEED and PathoLogic) and alternative gap filling methods: the gap-filling methods included in PathoLogic and SEED, the anabolic fast-gap filling, Bio-Growmatch and the combination of anabolic fast gap-filling followed by Bio-Growmatch. The gap-filling methods where, when possible generated in combination with the four biomass definitions. Additionally 4 manual models are included and a set of leave one out models are created. Each of these combination has been compared to the Biolog data processed with the OPM package from which confusion matrices and statistical estimators are calculated.

(FP) predictions. In the final step, a new consensus set of reactions is identified so that a minimal number of reactions are removed from the model. This scheme closely follows that of the method proposed in [223], however important differences have been introduced, especially in steps 1, where fastGapFill is used, and in step 3 that simultaneously considers all media for which experimental data are available. Additionally, steps 2 and 4 imply solving a Mixed integer linear programming (MILP) problem. In the Bio-Growmatch implementation, the MILP problem has been split to reduce the memory load and the computational cost. Running time varies depending on the model and on the size of the experimental data sets. As an example, running Bio-Growmatch with default parameter set using as input a model generated using the Seed algorithm and a dataset with 190 growth conditions, typically requires 2 to 6 hours running on 4 cores on a typical I5 core machine.

In addition to Bio-Growmatch we have developed an anabolic fast gap-filling method based on the already existing fastGapFill [289]. The main modification is a pre-defined minimal requirement for the objective function in our approach. This gap-filling approach is designed to optimize anabolic capabilities, meaning that it is assumed that the full set of catabolic reactions

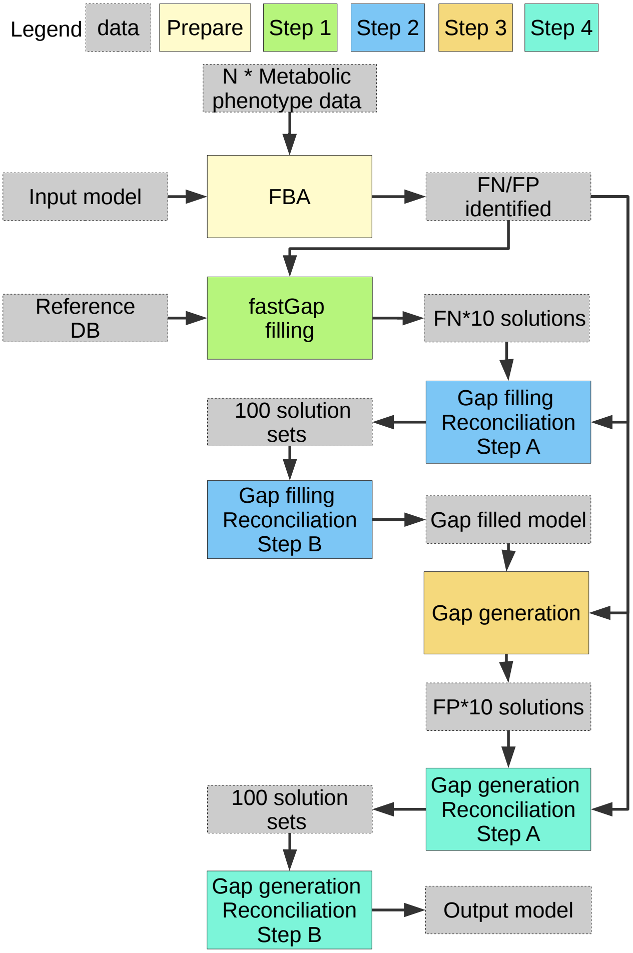


FIGURE 10.2: **Schematic overview of the steps in Bio-Growmatch.** Prepare) The prediction for each metabolite is determined and compared to the measured phenotype data. Step 1) For each FN, propose multiple gap filling solutions. Step 2) Reconsolidation I, select the solutions that solve most FN while minimizing the modifications to the model. Step 3) For each FP propose multiple gap generating solutions. Step 4) Reconsolidation II, select the solutions that solve most FP with the minimum amount of changes to model.

are present in the model. This is achieved by allowing, during the gap filling process, unlimited uptake of TCA cycle intermediates and closely related compounds: oxaloacetic acid, citrate, aconitate, isocitrate, α-ketoglutarate, succinate, fumarate, malate, pyruvate, glutamate, glutamine and aspartate. Bio-Growmatch and the anabolic fast gap-filling method have been implemented as a Java command line application and are freely available.

Definition of objective functions

Four objective functions were used in the gap-filling process and to make predictions with the models. Three of them represent alternative biomass compositions: The first one, which we termed 'Full' is based on the biomass definitions that can be found in MetaCyc [95] and ModelSEED [224] databases. The 'Basic' biomass definition requires cofactors ATP, NADH and S-adenosyl methionine together with aspartate, glutamate, serine, α-ketoglutarate, pyruvate and ribose 5-phosphate. The 'Medium' biomass function reports growth if the model can synthesize all metabolites at the main branch point of metabolism. The fourth objective function, which we termed 'Mini' is used to predict catabolic capabilities. Flux through this reaction indicates that the compound of interest can be degraded to any one of the TCA cycle intermediate metabolites. In the models this is represented as a synthesis reaction requiring either one of the compounds in the TCA cycle. A detailed list of the components in each objective function is provided in Additional file 1.

Metabolic phenotype of selected bacterial species

Biolog phenotype microarrays produce measurements on cellular respiration of distinct substrates and provide a characterization of the catabolic capacity of the cells [446]. Here, we used the microarrays to characterize the ability of 31 bacterial species to catabolize potential carbon sources. The selected bacterial species represent a collection of species spread across the kingdom of bacteria, and includes members from 6 phyla, 16 orders and 28 genera (see Additional file 2)

The biolog data are presented in Additional file 3. These results show the great metabolic diversity of the selected species. No single compound appears that can be used by all the selected species, in fact the most utilized compounds are fructose, glucose and glucosamine, which are all three utilized by only 19 of the 31 species.

Experimental characterizations were performed for 190 carbon based compounds. However, only for 127 of these degradation pathways can be found in the ModelSEED and MetaCyc databases. Among the remaining 63 compounds, over 66% were not present at all in one of the database. See Additional file 4 for detailed information.

Automatically generated models

For each of the selected species we used alternative pipelines for automatic reconstruction of metabolic models. We have tested two genome annotation pipelines (SAPP [273] and RAST[31]), two automatic model generation tools (SEED and PathoLogic) and five alternative gap filling methods: the gap-filling methods included in PathoLogic and Seed, anabolic fast-gap filling, Bio-Growmatch and the combination of anabolic fast gap-filling followed by Bio-Growmatch (see Figure 10.1). The gap-filling methods where, when possible generated in combination with the four biomass definitions. Overall, we generated 868 GEM, 28 for each of the considered bacterial species.

Seed models are more extensive (average of 3010 reactions and 3232 compounds) that those generated using PathoLogic (an average of 1326 reactions and 1474 compounds), as shown in Table 10.1. The anabolic fast gap-fill added, on average, 8.52 reactions to the models. Bio-Growmatch, added an average of 30.74 reactions and removed 13.36 reactions. The combination of Bio-Growmatch and the anabolic fast gap filling lead to slightly smaller models with less added reactions (25.46 on average) and more removed reactions (19.02). See Additional file 5 for detailed information on each species.

TABLE 10.1: Average number of reactions added and removed in each gap-filling approach.

model-build method	bio-mass definition	avg. added #reactions AFGF+	avg. added #reactions Bio-Growmatch	avg. rem. #reactions Bio-Growmatch	avg. added #reactions AFGF&Bio-Growmatch	avg. rem. #reactions AFGF&Bio-Growmatch
PT	mini	0	15.42	48.68	15.29	47.9
PT	basic	0.19	13.87	33	15.97	42.36
PT	medium	2.03	13.19	0.55	16.07	22.97
PT	full	12.87	20.42	0	15.87	3.32
seed	mini	0	33.84	17.65	34.36	17.58
seed	basic	0.55	31.07	7.03	34.84	10.84
seed	medium	5.48	34.65	0	35.13	6
seed	full	47.07	83.48	0	36.19	1.19

AFGF refers to anabolic fast gap-filling. "+" and "-" indicate added and removed reactions respectively. Models had been generated with each of the indicated tools (PT indicates PathoLogic) and gap-filling has been performed using the indicated objective functions. Data represent averages over the studied 31 species.

Assessment of automatically generated GEMs

We performed a systematic evaluation of the potential of the models to predict the metabolic capacity of each of the selected strains. For each model, predictions for the carbons sources were compared with experimental data. These comparisons were done only for the 78 compounds for which degradation pathways are present in both ModelSEED and MetaCyc databases. For each model we have calculated statistical performance estimators, such as precision, sensitivity, specificity and accuracy. Given the diversity of the metabolic potential of the selected species, we have also computed the Matthews correlation coefficient (MCC) which provides a balanced measure of the quality of the predictions regardless of the number of positives and negatives [326]. Table 10.2 contains averages over all the studied species. Detailed information on each species can be found in Additional file 6.

Bio-Growmatch uses phenotype data for model optimization and, as expected, it greatly increases the agreement between model predictions and experimental data. Therefore, we further evaluated the quality of the predicted models on data not used for model training. To this end, for each of the 31 organisms and each of the tested 78 compounds, a model was built on which the gaps were filled using Bio-Growmatch and the full set of experimental characterization of the organism except data corresponding to the considered compound. The so generated models were used to assess the quality of the predictions on that particular compound. Data corresponding to this unbiased approach were averaged over the considered species. This is an extremely demanding approach as it entails the construction of 2418 models and the overall process requires over 40000 hours of computational time. Thus, we have only applied this approach to models generated using the SEED model construction pipeline and the basic biomass definition. Table 10.2 shows the increased predictive power of this approach when compared with automatically generated models.

This strategy of leaving out one dataset on the model reconstruction step, also provides the framework to compare the performance of the automatically created models (after anabolic fast gap-filling and Bio-Growmatch) with that of already published models. We retrieved available models for *Escherichia coli*, *Bacillus subtilis*, *Pseudomonas putida* and *Salmonella typhimurium* [356, 362, 368, 496]. These models were used to make predictions on substrate utilization and the results were compared with the experimental data. The outcome of this analysis is summarized in Table 10.3.

Discussion

Comparison of model building methods, biomass definitions and gap-filling methods

Precision and sensitivity of the models generated without any additional gap-filling procedure are zero when the Full biomass definition is used (Table 10.2). This is due to the fact that for none of the tested species, the automatic reconstruction methods led to networks with complete pathways for synthesis of elementary biomass components such as DNA, RNA, lipids or proteins. This further emphasizes the need of efficient gap-filling methods for automatic model generation.

TABLE 10.2: Scores for the generated models

modelbuild method	**biomass definition**	**gapfill method**	**sensitivity**	**specificity**	**precision**	**accuracy**	**MCC**
seed	mini	none	0.55	0.67	0.49	0.63	0.22
seed	mini	default	0.56	0.67	0.48	0.63	0.22
seed	mini	AFGF	0.55	0.67	0.49	0.63	0.22
seed	mini	Bio-Growmatch	0.88	0.9	0.82	0.89	0.77
seed	mini	AFGF& Bio-Growmatch	0.89	0.9	0.83	0.89	0.78
seed	basic	none	0.39	0.8	0.52	0.65	0.2
seed	basic	default	0.45	0.78	0.53	0.66	0.24
seed	basic	AFGF	0.55	0.68	0.49	0.63	0.23
seed	basic	Bio-Growmatch	0.81	0.9	0.82	0.87	0.71
seed	basic	AFGF& Bio-Growmatch	0.81	0.93	0.86	0.89	0.75
seed	basic	leave one out	0.52	0.85	0.66	0.73	0.39
seed	medium	none	0	1	0	0.64	0
seed	medium	default	0	1	0	0.64	0
seed	medium	AFGF	0.55	0.68	0.49	0.63	0.23
seed	medium	Bio-Growmatch	0.75	0.85	0.73	0.81	0.59
seed	medium	AFGF& Bio-Growmatch	0.87	0.85	0.76	0.86	0.7
seed	full	none	0	1	0	0.64	0
seed	full	default	0	1	0	0.64	0
seed	full	AFGF	0.56	0.67	0.49	0.63	0.22
seed	full	Bio-Growmatch	0.76	0.84	0.73	0.81	0.6
seed	full	AFGF& Bio-Growmatch	0.9	0.77	0.68	0.82	0.64
PT	mini	none	0.84	0.29	0.4	0.49	0.15
PT	mini	default	0.85	0.29	0.4	0.49	0.15
PT	mini	AFGF	0.84	0.29	0.4	0.49	0.15
PT	mini	Bio-Growmatch	0.93	0.79	0.71	0.84	0.69
PT	mini	AFGF& Bio-Growmatch	0.94	0.79	0.71	0.84	0.69
PT	basic	none	0.66	0.43	0.39	0.51	0.09
PT	basic	default	0.67	0.42	0.39	0.51	0.09
PT	basic	AFGF	0.84	0.29	0.4	0.49	0.14
PT	basic	Bio-Growmatch	0.73	0.81	0.68	0.78	0.53
PT	basic	AFGF& Bio-Growmatch	0.79	0.83	0.72	0.82	0.61
PT	medium	none	0.1	0.9	0.36	0.62	0
PT	medium	default	0.14	0.85	0.34	0.6	-0.01
PT	medium	AFGF	0.84	0.29	0.4	0.49	0.14
PT	medium	Bio-Growmatch	0.62	0.79	0.62	0.73	0.41
PT	medium	AFGF& Bio-Growmatch	0.87	0.64	0.57	0.72	0.49
PT	full	none	0	1	0	0.64	0
PT	full	default	0	1	0	0.64	0
PT	full	AFGF	0.84	0.29	0.4	0.49	0.14
PT	full	Bio-Growmatch	0.58	0.8	0.62	0.72	0.39
PT	full	AFGF& Bio-Growmatch	0.96	0.41	0.47	0.61	0.39

Statistical estimators for each combination between the two model generating methods, four biomass definitions and four gap filling options combined over all 31 models. The score for the leave-one-out test is also included. AFGF refers to anabolic fast gap-filling and AFGF & Bio-Growmatch to its combination with Bio-Growmatch

TABLE 10.3: Comparison between manually curated and automatically generated models

methods	sensitivity	specificity	precision	accuracy	f1	MCC
seed & mini & TCA-fast gap fill & Bio-Growmatch	0.89	0.9	0.83	0.89	0.86	0.78
manual & mini	0.57	0.79	0.73	0.68	0.64	0.37
seed & full & TCA-fast gap fill &Bio-Growmatch	0.9	0.77	0.68	0.82	0.78	0.64
manual&self	0.14	0.96	0.79	0.56	0.24	0.18

For the automatically generated models the results are combined over 31 models. For the manual models the results are combined over 4 models.

Similarly, when the Medium biomass definition is used, the number of positive results is also zero for SEED generated models. However, models generated with PathoLogic show a positive, albeit small, number of positive predictions, which indicates a higher degree of model completion. Nevertheless, even without gap-filling the automated methods are able to, at least partly, include catabolic pathways as shown by the relatively high accuracies obtained when using the Mini objective function. This is objective function is, as previously stated, an indicator of the catabolic capacities of the organisms detached from their anabolic capacities.

Default gap-filling methods included in the tested algorithms for automated reconstruction result in an insubstantial improvement of the generated models, as shown in Table 10.2. This is most likely due to mismatches between the biomass definitions here presented and the approaches used by each of the algorithms.

Applying the anabolic fast gap-fill approach to the generated models leads to clear improvements in the predictive power of the models regarding the Medium and Full biomass objectives. MCC goes from zero up to ≈ 0.22 and ≈ 0.15 for models generated using SEED and PathoLogic respectively, regardless of the selected biomass definitions (Table 10.2). These improvements are not surprising, given that this approach is specifically designed to improve the predictive power of the models regarding the anabolic potential of the species. These improvements are achieved at the cost of including a relatively large amount of reactions for which no genomic evidence is available (Table 10.1). SEED generated models seem to outperform the PathoLogic generated ones after anabolic fast gap-fill. Nevertheless, it should be noted that the number of reactions that are added in the SEED generated models almost doubles the number of reactions the PathoLogic generated models require. The anabolic fast gap-fill method was not intended to increase the predictive power of the model regarding catabolism, and, as expected, no modifications are introduced in the model (Table 10.1)

The gap-filling methods embedded in the reconstruction algorithms and the anabolic fast gap-filling methods result in addition of reactions to the model, as these approaches work based on adding reactions that enable synthesis of biomass components. However, Bio-Growmatch aims at the maximization of the agreement between model predictions and measured phenotypes. As a result, reactions are both added and removed from the model (Table 10.1). The more complex the biomass definition is, the more reactions are added and less reactions are removed, regardless of the originating model. For the Full biomass definition no reactions are removed. The resulting models achieve MCC that are significantly better than the scores achieved with no gap-filling, the gap filling method included in Seed and PT, or the anabolic fast gap-fill method. In particular, Bio-Growmatch clearly improves the specificity of the models.

Finally, the best performing models arise as a result of the combination of Bio-Growmatch with the anabolic fast gap-filling method, which is not surprising given that this combination is the one that introduces the highest number of modifications (added and removed reactions, as shown in Table 10.1). Bio-Growmatch is able to improve the models by reducing the number of incorrect positive predictions. However its applicability is limited when the starting model has such a high number of gaps that it is unable to simulate biomass synthesis and no positive predictions can be generated. Thus initial applications of the anabolic fast gap-filling approach improves the MCC and sensitivity scores, however it does so at the cost of specificity (Table 10.2).

Overall, Table 10.2 shows that, regardless of the starting model, application of Bio-Growmatch leads to models that better describe the metabolic potential of the studied organisms. Moreover, it also increases the predictive power of the model. This can be seen by comparing rows SEED/basic/ngf, SEED/basic/Bio-Growmatch and SEED/basic/AFGF&Bio-Growmatch in Table 10.2.

The comparison between manually curated models and those generated by Bio-Growmatch shows the surprising result of Bio-Growmatch outperforming the manually curated ones. However, caution should be exerted when interpreting this result. Bio-Growmatch has been designed to generate models able to account for metabolic phenotype data, even though in the carried on comparison the corresponding data was purposely left out of the process. On the other hand, published models have undergone a curation process that often takes into account other data types, such as gene essentiality. Moreover, it should be noted that in the automatically generated models we have assumed the existence of transporters needed to uptake the corresponding compounds. Transport is often a major bottleneck in substrate utilization that in the gap-filling method has not been considered, whereas in manually curated models curation of transport mechanisms is often a major effort. Finally the score of the curated models can be hampered due to metabolite mapping issues. This is an intrinsic problem often found in the manual models, were unique and univoque metabolite identifiers for the metabolites, such as InChI identifiers [], are not included in the annotation.

Selection of the objective functions for gap-filling

The choice of the objective function used in the gap-filling process has an important impact on the outcome of the algorithm. The best statistical estimators are obtained when Bio-Growmatch is used (possibly in combination with the anabolic fast gap-filling) together with the Basic biomass or the Mini objective function (Table 10.2 and Additional file 7). This difference in performance is not surprising as the Mini objective function is specifically designed to account for catabolic capabilities. Biolog microarrays produce measurements on cellular respiration on distinct substrates. A positive result implies that the cell is able to degrade the selected compound but it is not always guaranteed that is additionally able to use it as a sole carbon source. Other approaches to experimentally determine substrate utilization, such as growth curves emphasize the role of the carbon source in anabolism. In those cases, the Basic biomass could provide models that better incorporates the experimental data.

Impact of the reference databases

The quality of the model enhancement achieved with the Bio-Growmatch depends on the quality of both the experimental data and of the reference database. However, alternative namespaces might be used for both, which might hamper their interlinking. This prevents correct identification of degradation pathways. Here, these problems have arisen for a quarter of the 190 carbon based compounds in the Biolog microarray, as either they could not be identified or there was no known degradation pathway in the reference database. These typically include the more complex molecules containing a linked set of sugars, amino acids and fatty acids. However, this issue also affects simple but more rare molecules such as lactulose as well as other frequently used compounds, such as Tween 40.

SEED and PathoLogic

PathoLogic requires annotated genomes with locus tags associated to GO terms, EC numbers, and enzyme names. This information is then used to identify the corresponding reactions in the MetaCyc database. However, SEED model building requires the genomes to be annotated using RAST which uses the ModelSEED database. The database used by PathoLogic currently contains 12399 balanced reactions and uses compound classes whereas the one used by SEED has 26079 balanced reactions and compound classes are not used. These might be the cause of the differences between the generated models, as models generated with PathoLogic show a higher sensitivity whereas the SEED generated ones result in higher selectivity. Those differences might also be the cause of the better performance of the SEED generated models after gap-filling using Bio-Growmatch.

Conclusion

Bio-Growmatch is able to improve automatically generated models so that they better represent the known physiology of the organism. Phenotype data collection is an essential pre-requisite for Bio-Growmatch application. Here we have used phenotype microarrays, as they provide high-throughput data that in the past have been proven useful to increase the quality of metabolic reconstructions [54]. Nevertheless other assay technologies like the API 20E kit can also provide this type of data in a high-throughput manner [227].

Functional genome annotation is another critical step, here we have seen that the combination of RAST annotation and SEED model building produces the best results, however, the SEED gap-filling step is not needed. We have used a generic biomass function 'Full' to account for the cellular components of the selected organisms. This is similar to the approach followed by PathoLogic and SEED, however, within Bio-Growmatch this definition can be modified to the species of interest if experimental characterization is available. This has the potential of greatly improving the resulting model [528]. Finally, both the anabolic fast gap-fill and Bio-Growmatch should be used in combination. The Mini objective function will result in a better representation of the catabolic capabilities, however, using the Full biomass as an objective function would increase the scope of the model.

Bio-Growmatch is able to automatically include metabolic phenotype characterizations in the model building process. This leads to model drafts that represent a better starting point for the curation process, thereby diminishing the amount of manual labour required to obtain accurate genome-scale constraint based metabolic models.

Availability

The Bio-Growmatch code is available under the MIT license and can be accessed at www.gitlab.com/wurssb/Bio-Growmatch. Bio-Growmatch is implemented in Java, uses the Gradle build system and Cplex linear problem solver for solving the MILP [62].

Materials and Methods

Bio-Growmatch

Bio-Growmatch is designed to simultaneously minimize the number of false negatives (FN) (no growth or no degradation of compounds predicted when growth or degradation is experimentally proven to occur) and false positives (FP) (growth or degradation of compounds predicted when no growth or degradation is experimentally proven to occur).

The algorithm consists of four steps, summarized in Figure 10.2.

Step 1. Propose multiple gap filling solutions

In this step, the algorithm tries to identify up to 10 unique solutions that would resolve each FN leading to a positive predictions. The method searches for solutions with a minimal set of weighted modifications. Modifications include addition of reactions from a reference database or making an existing reaction reversible. Modifications are weighted according to:

$$\lambda_{f_i} = 1 + P_{structure_i} + P_{known\Delta G_i} + P_{unfavorable_i}\left(3 + \frac{\Delta G_i}{10}\right).$$

$P_{structure_i}$ is 1 if one of the compounds within an added reaction has an unknown structure and 0 otherwise. Similarly $P_{known\Delta G_i}$ is 1 if the change in the Gibbs free energy of the reaction, ΔG_i, is unknown and 0 otherwise. The penalty introduced by $P_{unfavorable_i}$ scales with ΔG_i thus penalizing reactions in a thermodynamically unfavourable direction. If the change in the Gibbs free energy is unknown the variable ΔG_i is set to zero [223].

To improve runtime performance we applied a strategy similar to the fastGapFill method [289]. The goal is to maximizes the flux through the objective function while minimizing the weighted sum of the flux through any of the additional reactions from the reference database (equation 10.1):

$$maximize\ M(\overline{c} \cdot \overline{v}_m) - \overline{\lambda}_f \cdot \overline{v}_r. \tag{10.1}$$

The vector $\overline{v}_m$ represents the flux through the reactions already in the model and the Boolean vector $\overline{c}$ indicates which reaction is the objective function. $\overline{v}_r$ represents the flux through the added reactions. M is equal to the sum of all λ_{f_i} penalty scores and ensures that the flux through the objective function is maximized:

$$M = \overline{1} \cdot \overline{\lambda}_f. \tag{10.2}$$

The maximization is performed with the additional constraints:

$$S \cdot \overline{v} = \overline{0} \tag{10.3}$$

where

$$\overline{v} = [\overline{v}_m, \overline{v}_r] \tag{10.4}$$

and

$$\overline{0} \leq \overline{v} \leq \overline{v}_{ub}. \tag{10.5}$$

S is the stoichiometry matrix of all reactions in the model and equation 10.4 represents the steady state hypothesis (no net accumulation or consumption of metabolites). All fluxes are constrained to be positive but below the flux upper bound $\overline{v}_{ub}$ (Eq. 10.5). For equation 10.5 to hold the model has to contain only irreversible reactions, this requires all reversible reactions to be split in a forward and a reverse reaction. Integer cutting is used to find multiple solutions [].

Step 2 Reconsolidating I

Step 2 selects the best set among the previously identified solutions. The best set of solutions is required to resolve the maximum amount of FN while minimizing the number of newly introduced FP. A secondary requirement is added to minimize the amount of weighted modifications to the model.

We closely follow the approach of Modified GrowMatch [] however, here the original MILP problem is split and solved in two steps, 2A and 2B, independent of each other. This is needed to overcome performance and memory issues that arise due to the increased size of the problem in the application of Bio-Growmatch.

Step 2A uses integer cuts to find up to 100 solution sets. Each solution set contains solutions from Step 1 and is used to identify the ones with the lowest FN and FP.

$$maximize\ Q\left(\overline{1} \cdot \overline{o}_{fn} + \overline{1} \cdot \overline{o}_{tn}\right) - \overline{1} \cdot \overline{s}, \tag{10.6}$$

where: $\overline{1} \cdot \overline{o}_{fn}$ is the number of corrected FN ; $\overline{1} \cdot \overline{o}_{tn}$ is number of preserved true negatives (TN) ; The Boolean vector $\overline{s}$ indicates, for each solution, whether it is to be included in the final set of solutions and $Q = n_{sol} + 1$ with n_{sol} the number of solutions.

Additionally the following constraints are imposed that take into account the total set of $\overline{n}_{fn}$ FN before the application of Bio-Growmatch:

$$\overline{o}_{fn_k} \geq \overline{\epsilon}_k \cdot \overline{s} \quad with \quad k \in \{1, \ldots, n_{fn}\} \tag{10.7}$$

and

$$n_{sol} * (1 - \overline{o}_{tn_k}) \geq \left(\overline{1} - \overline{\mu}_k\right) \cdot \overline{s} \quad withk \quad \in \{1, \ldots, n_{tn}\}. \tag{10.8}$$

Here $\overline{o}_{fn_k}$ is equal to 1 if the k^{th} FN is turned into a TP by the addition of the reactions in the solution set and 0 otherwise. Similarly, $\overline{o}_{tn_k}$ equals 1 if the k^{th} TN, is preserved and 0 otherwise. For each FN, the vector $\overline{\epsilon}_k$ contains, for each solution considered, 1 if the solution gives a corrected prediction and 0 otherwise. $\overline{s}$ is a vector of size equal or smaller than $10 * n_{fn}$, that contains sets of reactions. The Boolean vector $\overline{\mu}_k$ contains, for each solution $\overline{s}_j$ and for each compound causing the l^{th} TN, 1 if the model with the inclusion of solution,$\overline{s}_j$, still gives a correct prediction and 0 otherwise. Equation 10.7 ensures that for each FN at least one solution turns it into a TP. For each preserved TN none of the solutions should give a positive solution (equation 10.8).

Step 2B selects the solution set with the minimum amount of weighted changes to model:

$$minimize\ \overline{\lambda}_f \cdot \overline{y}, \tag{10.9}$$

where $\overline{y}$ contains, for each reaction, 1 if the reaction is included the final model and 0 otherwise.

For each solution, j, in the solution set, all reactions have to be included in the model:

$$\overline{\delta}_j \cdot \overline{y} \geq \overline{s}_j \left(\overline{\delta}_j \cdot \overline{1}\right) \quad with \quad j \in \{1, dots, n_{sol}\}, \tag{10.10}$$

where $\overline{s}_j$ equals 1 if the solution j is included in the model and 0 otherwise and $\overline{\delta_j}$ contains for each reaction a 1 if the reaction is included in solution j and 0 otherwise. Each of the added reactions are marked such that they will not be removed again in the subsequent gap generating step.

Step 3 Propose multiple gap generating solutions

In this step the algorithm tries to find up to 10 unique solutions to fix each FP. A solution contains a minimal set of weighted modifications required to get a negative growth prediction. Modifications include the removal of a (set of) reactions or making a reaction irreversible.

Modifications are weighted with:

$$\lambda_{g_i} = 1 + P_{irreversible_i} \tag{10.11}$$

$P_{irreversible_i}$ imposes a penalty of 1 if the reverse reaction is not present in the model and of 0 otherwise. Thus, removing a reversible reaction has a greater penalty than making it irreversible.

To fix a FP associated to a given compound, a search is done for a solution predicting no growth on the compound while predicting growth for all the compound associated to each existing TP. To do so, first an FBA optimization is performed on which exchange reactions for all compounds associated to each TP are added. Subsequently, the obtained value of the objective function is set as minimum value (α) for the objective function in the bi-level optimization problem described below.

$$maximize\ \overline{\lambda} \cdot \overline{y} \tag{10.12}$$

subject to:

$$maximize\ \overline{c} \cdot \overline{v_m} \tag{10.13}$$

inner subject to:

$$S \cdot \overline{v}_m = \overline{0} \tag{10.14}$$

$$\overline{0} \leq \overline{v}_m \leq \overline{y} \cdot \overline{v}_{ub} \tag{10.15}$$

$$\overline{c} \cdot \overline{v}_m = \overline{0} \tag{10.16}$$

$$\overline{0} \leq \overline{t}_m \leq \overline{y} \cdot \overline{v}_{ub} \tag{10.17}$$

$$\overline{k} \cdot \overline{y} = \overline{1} \tag{10.18}$$

$$\overline{r} \cdot \overline{t} = 0 \tag{10.19}$$

$$\overline{t} = \left[\overline{t}_m,\ \overline{t}_e\right] \tag{10.20}$$

$$S \cdot \overline{t} = \overline{0} \tag{10.21}$$

$$\overline{t}_e = e_{influx} * \overline{1} \tag{10.22}$$

$$\overline{c} \cdot \overline{t} \geq \alpha \tag{10.23}$$

$$\overline{1} \cdot (\overline{1} - \overline{y}) \leq \omega. \tag{10.24}$$

Were: $\overline{\lambda}$ contains for each reaction present in $\overline{v}_m$ the weighting factor given in formula 10.11. The vector $\overline{y}$ contains, for each reaction in $\overline{v}_m$, a 0 if its removed in the final solution and 1 otherwise. The Boolean vector $\overline{c}$ indicates which reaction is the objective function. $\overline{v}_m$ contains the fluxes of all the reactions in the model including the exchange reactions and objective function. $\overline{t}_m$ is the counterpart of the $\overline{v}_m$ in the inner problem. The vector t_e contains the fluxes through the exchange reactions representing uptake of compounds associated to TP. $\overline{v}_{ub}$ contains the upper bounds for fluxes through each reaction in the model. The vector $\overline{k}$ is the selector for those reactions that should not be removed. The vector $\overline{r}$ is the selector of the exchange reaction associated to the processed FP. S is the stoichiometry matrix. e_{influx} contains the upper bounds for each exchange reaction. α equals the minimum flux through the objective function that should be achieved. The constant ω is equal to the maximum number of reactions that can be removed.

In the original "modified GrowMatch" method [223] the gap generating step is based on a bi-level optimization problem, which must find a solution of reactions $\overline{y}$ to remove such that the model with the knockout included (represented by the fluxes in vector $\overline{v}_m$) no longer predicts growth while the complete model (represented by the fluxes in vector $\overline{t}$) keeps predicting growth. The inner problem represents the optimization of $\overline{v}_m$ for maximal growth. In our approach, both the inner and outer problems represent the complete model, described by $\overline{v}_m$ and $\overline{t}_m$. However, within the inner problem the model utilizes the compound associated to the processed FP. Whereas in the outer problem the model is forced to utilize all compounds associated to TP for growth.

The gap generating step maximizes the (weighted) number of reactions kept in the model thereby minimizing the (weighted) modifications (equation 10.12). The maximal possible flux through the objective function of the model consuming the compound associated to the FP should be zero (equations 10.13 and 10.16). Equations 10.14 and 10.21 represent the steady state hypothesis in the inner and outer problems respectively. All fluxes are constrained to be positive but below the flux upper bound constant $\overline{v}_{ub}$ (eq. 10.17). For equation 10.17 to hold the model has to contain only irreversible reactions, this requires all reversible reactions to be split in a forward and a reverse reaction. Transport and biochemical reactions are within the model can be a candidate to be removed, and their removal is represented in the Boolean vector $\overline{y}$ (eq. 10.17). However, exchange reactions, the objective function, the influx reaction for the compound associated to the FP and reactions added in the gap filling step, represented in vector $\overline{k}$, are excluded from the candidate for removal list (eq. 10.18). Uptake of the compound associated to the processed FP must be zero (eq. 10.19). The model in the outer problem includes the influx reactions for all the compounds associated to each of the TP (eq. 10.20). To mimic a forced growth on the compounds associated to each TP, each of the influx reactions should have flux and the objective function should have a minimal flux equal to the combined flux possible for all compounds consumed (eq. 10.22 and 10.23). The minimum biomass production needed (α) is equal to the flux achieved in a pre executed FBA, in which the biomass production was maximized and in which all compounds associated to each TP are included. This ensures that the model must be able to simulate grow on each of the compounds, instead of simulating their degradation through respiration or fermentation. Finally, to ensure a reasonable computational time the number of reactions to remove is limited to a preset constant ω, with default value to 3 (eq. 10.24).

Step 4 Reconsolidating II

This is the same as step 2 (described in eqs. 10.6 to 10.10 , with the exception that the $overline\lambda_f$ weighting factor is replaced with the $\overline{\lambda}_g$ factor, the elements reading FN, TN, TP and FP should read FP, TP, TN and FN respectively (that is positives and negatives exchage their roles) and the vector $\overline{y}$ contains for each reaction a 1 if the reaction is excluded from the final model and 0 otherwise.

Anabolic fast gap-fill

The developed anabolic fast gap-filling method is based on the already existing fastGapFill [495]. For the anabolic fast gap fill, the model is modified to only contain irreversible reaction. This again requires all reversible reactions to be splitted in a forward and a reverse reaction. Then, the following linear problem is posed:

$$minimize\ \overline{1} \cdot \overline{v}_r, \tag{10.25}$$

subject to:

$$\overline{v} = [\overline{v}_m, \overline{v}_r] \tag{10.26}$$

$$S \cdot \overline{v} = \overline{0} \tag{10.27}$$

$$\overline{0} \leq \overline{v} \leq \overline{v}_{ub} \tag{10.28}$$

$$\overline{c} \cdot \overline{v}_m > \alpha. \tag{10.29}$$

Here $\overline{v}_r$ is the set of irreversible reactions in the reference database. The vector $\overline{v}_m$ is the set of reactions in the model. The Boolean vector $\overline{c}$ indicates which reaction is the objective function and α is the pre-defined minimal requirement for the objective function. Thus, the flux through the reactions added from the reference database is minimized while ensuring a minimal value ($\alpha = 0.005$) for the objective function.

In the anabolic fast gap-fill exchange reactions are included (or modified if existing) to allow uptake of TCA cycle intermediates and closely related compounds: oxaloacetic acid, citrate, aconitate, isocitrate, α-ketoglutarate, succinate, fumarate, malate, pyruvate, glutamate, glutamine and aspartate.

Data preparation

Biolog data was generated for the 31 species listed in additional file 1 using the BIOLOG PM01, and PM02A microplates containing carbon sources. The protocol described in [54] was followed. In brief, bacteria were grown overnight on nutrient agar plates. Biolog experiments were performed according to the modified protocol "PM Procedures for *E. coli* and other GN Bacteria" (Biolog, Inc. 16 Jan 2006; see Supporting Information). Subsequently, cells were transferred and suspended into 20 ml of Inoculating Fluid IF-0 to achieve 85% T (transmittance) in the BIOLOG Turbidimeter. About 240 μl Dye Mix A and 3760 μl H_2O were added to a final volume of 24 ml. Wells wells were inoculated with 100 μl of the 85% T cell suspension. Experiments were carried in triplicate.

Data was processed with both the OPM[506] and RBiolog[511] R packages using default parameters. OPM produces a result for each replicate whereas RBiolog also provides a summarized result. In case of conflicting results, 'growth detected' was selected. Both methods produce comparable results, however, the OPM method reports more positives. We tested the two methods with the generated models and found a slightly better result for OPM. Upon closer investigation we noticed that RBiolog is likely to misclassify a positive growth result if there is a delayed growth curve. Therefore OPM was selected. Raw and processed results are provided in additional file 3.

Automated GEM reconstruction

For each of the 31 species we generated 14 automatic models using PathoLogic and 14 using SEED. An overview of the generated models is provided in Figure 10.1. Models built with PathoLogic were based on the functional genome annotation generated using the Prodigal, InterProScan, Swissprot BLAST and PRIAM modules from the SAPP framework [273] whereas the ones created with SEED were based on the RAST genome annotation with default parameters. Alpha-D-glucose was used as a carbon source in the optional gap filling performed in SEED.

SMBL versions of the GEM models for *E. coli* (JO1366 [368]), *B. subtilis* (iYO844 [362]), *P. putida* (iJN746 [356]) and *S. typhimurium* (STM_v1_0 [496]) were download from BiGG database [269]. All these models are in the BiGG namespace.

Reference databases

MetaCyc, the reference database from PathoLogic was retrieved. The SEED reference database files with the reactions *reactions.master.tsv* and compound definition file *compounds.master.tsv* were retrieved from `https://github.com/ModelSEED/ModelSEEDDatabase/tree/2d8a6afb0929a8b8ea21a164de89e36bf2c04367/Biochemistry`. Direction directionality was updated with information from the file *Reactions.tsv* downloaded from `https:`

//github.com/ModelSEED/ModelSEEDDatabase/tree/2d8a6afb0929a8b8ea21a164de89e36bf2c04367/Templates/Core/. Stoichiometrically unbalanced reactions were removed from the reference databases.

Reference databases were interlinked. MetaCyc and SEED were linked using InChI identifiers, whereas the internal cross links to the MetaCyc and SEED identifiers were used to map BiGG identifiers. MetaCyc includes compound classes and some compounds in the Biolog PM01, PM02A Microplates link to these classes. For these we added a mapping linking the compound class to each of its instances.

Biomass definitions and exchange reactions

The Full biomass definition is based on the biomass definition of *E. coli* from MetaCyc [521] and the SEED biomass definitions available under https://github.com/ModelSEED/ModelSEEDDatabase/tree/f92036b50c503e9ab950bfc6ac75f18a39213e3d/Templates. Only compounds present in both reference databases were kept. To overcome the mismatch caused by the missing link between the instances of fatty acids and the general class 'Long-Chain-Aldehydes' we added this class to the inward flow exchange reactions set for the PT variant.

Exchange reactions for water, ammonium, sulfate, phosphate, protons, chlorine, bromine and biological relevant metals have been included in the models to simulate free in and out ward flow. Oxygen and carbon dioxide were included in the inward and outward exchange reaction set respectively. A curated mapping of all the compounds used in the exchange reactions, biomass definitions and Biolog microplates can be found in additional files 1 and 3.

Model simulations

FBA was used to find the maximum flux value through the selected objective functions (defined in additional file 1). A simulation was considered to produce a positive prediction when this maximal flux was bigger than the preset cutoff (0.005). Exchange reactions representing the media are defined in additional file 1. Additionally an inward exchange reaction was added to simulate cytosolic availability of the tested compound

To assess model quality we built confusion matrices with the results of the comparisons of model predictions (additional file 7) to experimental data (additional file 3) for the 78 tested compounds (that are indicated in additional file 3). From these matrices we calculated, using standard definitions, sensitivity, specificity, precision and accuracy, we also computed Matthews correlation coefficient (MCC), defined [326]:

$$MCC = \frac{TP \times TN - FP \times FN}{\sqrt{(TP+FP)(TP+FN)(TN+FP)(TN+FN)}}, \qquad (10.30)$$

where TP indicates the number of true positives, TN of true negatives, FP of false positives, and FN of false negatives.

Acknowledgements

This work was supported by the European Union's 7th Framework projects Microme (Project reference: 222886). This work was carried out on the Dutch national e-infrastructure with the support of SURF Foundation.

Additional information and files

Electronic supplementary material can be accessed at www.gitlab.com/wurssb/Bio-Growmatch **Additional file 1:** Objective functions and exchange reactions

Definition of objective functions and exchange reactions used in simulations. Metabolites have been defined in the MetaCyc, ModelSEED and BiGG namespaces.

Additional file 2: Species list

Species used for the analysis. Species name, NCBI taxonomic identifier, ENA accession number and curated model name (if present are) given.

Additional file 3: The biolog Data. (This file is available upon request to jesse@jessevandam.nl)

The raw biolog data for 31 species for the two carbon plates after it has been processed by the software delivered by Biolog company and processed data with OPM. Species are identified by their NCBI taxonomic identifier. The 78 compounds used for the comparison are marked.

Additional file 4: Degradation pathways reference databases

Presence of the 190 tested compounds in MetaCyc or ModelSEED databases. Found: indicates compounds in the database for which a degradation pathway is present. Unconnected: the compound is found in the database but no degradation pathway is present. Not found: the compound is not described in the database.

Additional file 5: Number of reactions added in removed in each gap-filling approach for each species

AFGF refers to anabolic fast gap-filling. Models had been generated with each of the indicated tools (PT indicates PathoLogic) and gap-filling has been performed using the indicated objective functions. Species are identified by their NCBI taxonomic identifier.

Additional file 6: Statistical estimators for the generated models per species

The statistical estimator scores per species for each combination between the two model generating methods, four biomass definitions, five gap filling options averaged over all 78 compounds.

Additional file 7: Model predictions

The model predictions and Biolog results. Model predictions are included for each of the 31 species, each of the 78 compounds and each of the tested combinations.

Chapter 11

General Discussion

The goal of this thesis was to improve the prediction of genotype to phenotype associations with a focus on metabolic phenotypes of prokaryotes. This goal was achieved through data integration, which in turn required the development of supporting solutions based on semantic web technologies.

Objective and solutions

In my thesis I developed four sets of tools. The first set of tools (DIVA and SyNDI) is dedicated to the visualization and analysis of heterogeneous data sets with a focus on concurrent network view. DIVA was a prototype of SyNDI, which is implemented in the popular network viewing tool Cytoscape. DIVA and SyNDI are best suited for discovering novel genes involved in some biological process, function of genes and processes and analyzing (cross) regulation. The use of these tools was illustrated in chapters 2 3 and 4 where these tools were extensively used to unravel different aspects of gene regulation in *M. tuberculosis*. However these tools are less suitable to analyze metabolism.

GBOL is an ontology for genome annotations. Together with GBOL I have presented an associated genome annotation pipeline SAPP. This is a tool with a special focus on provenance and it is able to capture both historical and contextual provenance of the annotation. SAPP produces FAIR genome annotation data that can be linked with existing databases. FAIR genome annotation ensures that all annotations and associated contextual provenance (which includes the p values) are comparable and easy to query. This was vital for **chapter 9**.

To support the development of GBOL and SAPP we developed RDF2Graph and Empusa. RDF2Graph is a tool to recover the structure of an RDF resource. This can be used for understanding a resource and to know how to query it. I used RDF2Graph to recover the structure of the UniProt database and subsequently query it. Furthermore, this tool can be used to validate that all data is correctly encoded. So if one creates a tool that generates RDF one can check that the results are correct. We used it to validate the data we exported in the initial prototype versions of SAPP. Empusa can be used to define an ontology and create an associated data accessing API. This API performs a data consistency check and can be used to export RDF data. This was used in the latest versions of SAPP, greatly enhancing its quality and shortening the development time needed.

BioGrowmatch was developed to attain the final goal of this thesis: to predict associations between genotypes and phenotypes. Models of metabolism were used for this task. BioGrowmatch integrates phenotype growth data in the model building process. This is a method to gap fill a model such that the model better predicts the phenotype. The method was tested with a collection of datasets generated for that purpose.

During my thesis a number of issues arose related to data reuse and integration, provenance and existing databases, that will be discussed below. The developed approaches will contribute to introduce dry-lab cycles in systems biology and I will provide and example of such a cycle in genome annotation and metabolic modeling and I will discuss the role of semantic technologies in the implementation of such cycle. I believe that Empusa will greatly facilitate further advances. In the final sections I will discuss some of the current challenges in GEM building that remain to be solved.

Data reuse & integration

Data reuse and data integration are two different yet strongly interlinked aspects. Data integration within one study does not require data reuse, however, cross-study data integration requires reusing data sets from different resources. On the other hand, it is not necessary to have support for data integration for data set reuse. However, the likelihood that a data set will be reused will increase if the data can be easily integrated. This relationship is reflected in the "FAIR" acronym, that relates to a set of standards to support data reuse and data integration. The first two letters stand for *Findable* and *Accessible*, which are essential for data set re-use. The "I" stands for *Interoperable*, which indicates that a computer can interpret the data, so that it can be automatically combined with other data. The "R" stands for *Reusable*, which states "(meta)data meet domain-relevant community standards" [524]. This I interpret to mean that the data is compatible with standard tools used within the community, whereas *Interoperable* means data can be directly integrated, so that it can be queried within and across other resources in the field.

Solutions such as FAIRDOM focus on reuse of (raw) data files arising from experiments performed to address specific questions. These can be, for example, data from a set of experiments used to prove a dynamic model. Tools developed in FAIRDOM ensure that the data sets are findable and accessible through a searchable web interface. The tools also ensure that data are associated to the meta-data needed for data reuse. Associated to this data reuse efforts are meta data standardization efforts, such as MIMS, that describe sample origin, for example from where a given soil sample was collected [534].

Instead, other initiatives such as STRING and solutions as the ones presented in **chapters 2** and **3**, are devoted to creating fully processed data sets that are turned into information sets that can be readily integrated, which I will call pre-integrative information sets. In these solutions, the information sets are presented as networks that can readily be compared with each other in an easy-to-use interface. Similarly, our efforts in **chapters 5, 6 and 8** were devoted to fully processing the data to generate information sets that can be readily queried with SPARQL. The output of these queries, can be further processed using commonly applied tools such as R, as shown on **chapter 9**.

In my opinion, solutions to make data sets *Findable* and *Accessible* in public repositories are critical to enable creation of pre-integrative information sets. However, data available in public repositories, require a lot of work and bio-informatics expertise so that they can be processed and integrated into information sets that can be used to address biological questions. Moreover, the input of computational non-expert researchers is often critical to further interpret the pre-integrative information sets into new knowledge. I believe pre-integrative information sets to be more important than large data collections to address biological questions. The biology expert can use the former to enhance the discovery of new insights and the design of new experiments, as these kind of resources can be used to address the problems within a limited time span.

Pre-integrative information sets are most usable when they are easy to browse, query and integrate. One example of a resource with these characteristics is the UniProt database [50]. A website can be used to *browse* the information. A user can, for instance, browse on the website to find known 3D structures and information on single point mutations altering the function of a protein on interest. There is an additional interface to *query* the information. For example, to identify proteins that could potentially work as a light sensitive activator, the resource can be queried to find all proteins with annotated ATPase activity and at least one light sensitive protein domain. Finally, there is an RDF representation of the data, so that is can be readily *integrated* with other resources. In this way, to find possible regulatory mechanisms in a pathway of interest, Reactome [160] can be queried to identify all proteins known to be present in the pathway; the output can be cross queried in UniProt to select all protein that also have a phosphorylation domain present.

Tools for creating such pre-integrative information sets were presented in **chapters 6**, and **7**. In **chapter 6** we presented the GBOL stack, which enables to transform genome annotation into a format which supports data querying and integration. The results of the different annotation tools can be queried with a single SPARQL query that combines the results of multiple annotation tools. For example, all EC numbers predicted by BLAST, InterProScan, PRIAM and EnzDP [354] can be retrieved and combined. The development of GBOL was strongly supported by the Empusa code generator (presented in **chapter 8**). In **chapter 7** we presented SAPP, that can be used to process each genome sequence (data set) into a fully annotated genome(pre-integrative information set). We subsequently applied SAPP to a large number of available genome sequences, within the *Pseudomonas* genus. This resulted in a resource that can be readily queried and integrated with other sources, as shown on **chapter 9**. More over, features and possibilities offered by SAPP have been successfully applied to support the findings presented in more than twelve publications, see `http://sapp.gitlab.io/` for a list of publications on which SAPP has been used. Currently, in the Laboratory of Systems and Synthetic Biology a large number of over 100 000 bacterial genomes have been re-annotated using SAPP. Current efforts in the laboratory are directed towards the development of a browseable web-interface for these information sets and on the development of protocols for mining the data.

As clearly shown in **chapters 2, 6** and **9**, to create pre-integrative information sets, one must process and normalize the data sets with the same methods. Otherwise, comparison between the information sets becomes impossible, as the differences would be based on the methods used rather than on biological differences in the samples related to the data sets. However, for some measuring techniques, it is hard or impossible to normalize the data, so that data sets can be compared. For example, when two different types of gas chromatography columns are used in a GC-MS based experiment. Therefore, large data sets are better to create pre-integrative information sets than many small data sets created for specific research purposes. One example of such large data set is the Mtb expression data collection [72] that covers over more than 200 conditions, has been cited over 500 times (as of November 2017) and was extensively used in **chapters 2 and 3**.

Currently, large amounts of data sets with few samples have been generated, whereas a relative low number of samples can be found in large data sets (100+ samples), as indicated in Figure 11.1. I strongly believe that community efforts should be more devoted to the creation of large reusable data sets. Large pre-integrative data sets contribute to generate predictions and hypothesis that still need experimental proof. Experiments to prove this hypothesis can be more precise and targeted using the knowledge gained trough the use of pre-integrative data sets and therefore the hypothesis can be proved with less effort and cost. Therefore, funding agencies could encourage the generation this type of data sets instead of granting research in which data is only created to research a single question.

Provenance

Provenance holds the information that describes the processes that are responsible for the creation of a data set. This we hereby define as historical provenance (dataset-wise). Furthermore, we also define the additional information generated by these processes that give support to each of the results in the data set as contextual provenance (element-wise). This typically includes p values or other scores.

Provenance is critical for data reuse. Typically, a shared data set is associated only to the historical provenance, which can be used to find, to review and to correctly interpret the results captured within the data set. However, a large pre-integrative information set contains a (large) set of individual predictions, for which it is important to access the supporting evidence so that significance of the results can be put into context (contextual provenance). It

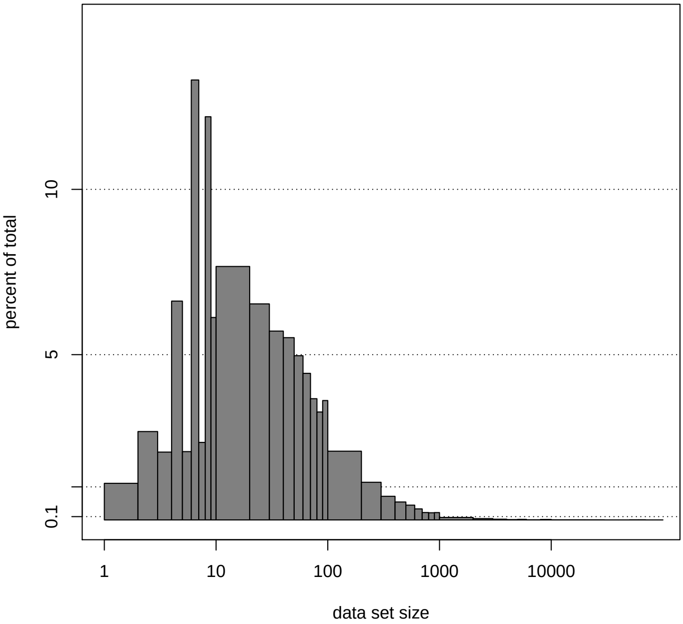


FIGURE 11.1: **Size of published transcriptomics data sets in the BioSamples database [203].** Distribution of the amount of samples included in each published data set. The y-axis represents the percentage of published data sets with a given number of samples (indicated on the x-axis). Note that the number of samples have been binned binned into bins 1 to 10 with bin size 1,10 to 100 with bin size 10, 100 to 1000 with bin size 100, and 1000 to 10000 with bin size 1000. The first 10 bins show a bias towards an even number of samples with a data set.

is this contextual provenance that is often lacking or incomplete. For example, in the TrEMBL data set [35] it is not possible to find significance values, although an evidence code to indicate the source of the evidence is present. Moreover, in this data set also the historical provenance is incomplete, as it impossible to find how the information was generated. Due to this issues, it becomes difficult or even impossible for a researcher to evaluate the value of the predictions in the TrEMBL data set.

Contextual information is often essential to correctly interpret computational outcomes. We used SAPP (described in **chapter 7**) to annotate, using the InterProScan module, a large number of bacterial genomes (>85.000). We evaluated the predicted occurrences of the PurE domain (IPR000031) and associated E-values (part of the contextual information). PurE is associated to the conversion of 5-aminoimidazole ribonucleotide (AIR) to 4-carboxy-AIR (CAIR) in the purine biosynthesis pathway. We used the intrinsic InterProScan threshold value (E^{-25}), which is derived, in this case, from the one in the Gene3D database [285]. As can be seen in Figure 11.2 A, the E-values associated to these instances show a bimodal distribution. These two peaks can be associated to two divergent paths in the *de novo* purine biosynthesis pathway. Conversion of AIR to CAIR in most bacteria proceeds through two enzymes (PurK and class I PurE) as indicated in Figure 11.2 B. However, class II PurE catalyzes this reaction in a single step. These two mechanisms can be associated to each of the peaks shown in the figure. Inspection of the identified hits shows that they overlap with instances of the domain signatures IPR033747 and IPR033626 from InterPro. These correspond to class I and II PurE, respectively. Class II PurE is present in animals, but it has also been found in a reduced number of bacteria [171]. This example shows how element-wise provenance can be used to provide additional context to individual findings and to uncover biological variants.

To capture the information needed to review and contextualize annotation information, provenance should, in my opinion, contain sufficient historical and contextual provenance data.

The historical provenance should contain, for each step, the following minimal information requirements: i) The input data sets and parameters used. For each input data source specific information should be available. For external files, historical provenance should include from where it was downloaded, when it was accessed and the version number of the data set, if available. For intermediate or temporary files, it should include a historical provenance of its own. For self-created data files, it should include from where and under which conditions the associated sample material was collected and it should contain which experimental measuring and data generation methods were used. For example "we collected a soil sample from a salt lake in a location with a given set of conditions and we used Sanger sequencing to sequence genomic DNA following the protocol as given in reference 'x'". ii) The start and end time of the job. This should be included whenever the code used to process the data set contains a call to a web service. iii) The code and associated version used.

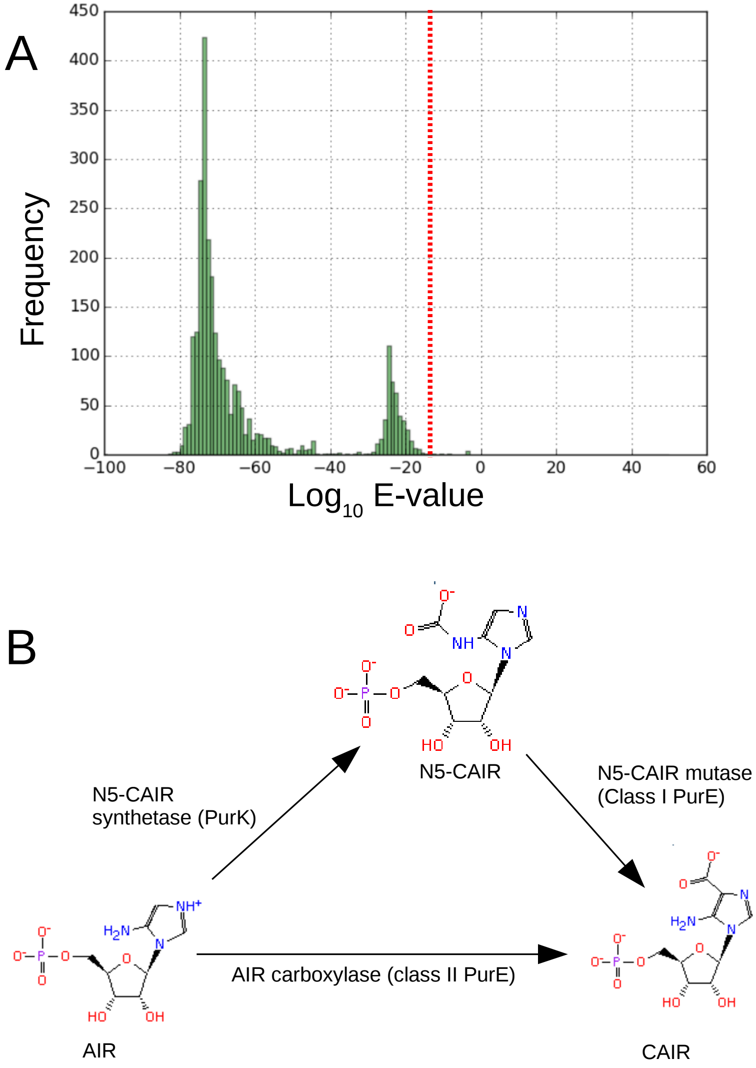


FIGURE 11.2: **PurE domain (IPR000031)**. **A)** Frequency distribution of the E-values associated to instances of PurE domain (IPR000031) identified using InterProScan. The bimodal nature of this distributions is apparent in the two shown peaks. N5-carboxy-AIR.**B)** Reactions from the de novo purine biosynthesis pathway converting 5-aminoimidazole ribonucleotide (AIR) to 4-carboxy-AIR (CAIR). N5-CAIR indicates the intermediate N5-carboxy-AIR.

A reference to the code repository (for example a git repository) must be included together with an identifier of the last commit that was applied to the version used to process the data set. If the data processing consists of multiple steps that can be contained in a single script, then that script should be put into a code repository and the provenance should only point to that script. This will ensure that the provenance does not become overly complex as the script will contain the information needed to know how the data came to be.

The contextual provenance should contain, for each individual prediction, information on to which data set it belongs, so that through this data set the historical provenance is linked. For computer generated predictions, contextual provenance should include significance scores and any other score related to the statistical framework of the used method. For example an E-value and coverage value for a BLAST based prediction.

Additionally, provenance on manual curation should also be included. Here, I define *manual curation* as a curation process based on human assessment. This process modifies the validity of a knowledge element. For manually curated annotations, contextual provenance should contain: i) Reference to a publication with results supporting the prediction. ii) Type of evidence, indicated with an evidence code from the evidence ontology [107]. iii) A textual note in English describing the reasons for the manual annotation. iv) Identification of the curator, with an ORCID [89], full name and optionally an organization. v) The date. vi) A mark (prove or negate) that indicates whether the added information in the curation helps to prove or to negate the statements to which the curation applies.

There are additional requirements when the annotation links elements from a different source such as a protein and an EC number. The link should be encapsulated into a cross reference to which the historical provenance, the element wise provenance, the source data set and target data set should be linked. Thus, it becomes possible to select all links between two data sets related to a given set of evidence. For example, a query can be written to retrieve all EC numbers linked to proteins in a genome of interest so that EC numbers are assigned based on sequence similarity (BLAST hit) to the Swiss-Prot database with an E-value lower than a given cutoff/bit score or alignment length.

These requirements have all been implemented into the GBOL ontology, described in **chapter 6**, except for the option to describe self created data sets. This is because we initially focused on the re-annotation of existing genomes, but it will be included in a follow up versions. Representation of historical provenance in GBOL is based on the preexisting PROV-O ontology [291], whereas the contextual provenance is based on a self created schema. PROV-O perfectly applies to the historical provenance, however fails to properly encode the contextual provenance. The SAPP annotation pipeline, presented in **chapter 7** generates all the information in these requirements.

The listed requirements for provenance of genome annotation data would provide researchers enough information to critically review existing results, thus ensuring that scientific results are cumulative and reproducible.

Existing Knowledge databases

In addition to large pre-integrative information resources like STRING and TrEMBL, other resources like Swiss-Prot, MetaCyc and InterPro exist. These result from long lasting manual curation efforts. These resources are ultimately based on integration of many different experiments and publications and can be seen as a pre-integrative information sets, which were turned into a knowledge-base. Within these resources supporting evidence for the statements is included, which mostly points to literature. However, the completeness, quality and especially standardization of the provenance of the original findings and experiments described in the papers are less then what can be achieved with computer generated knowledge stored in semantic web based stores. Furthermore, despite the peer reviewing process errors might have accumulated.

To regain the high quality provenance track, we should complement these resources with automatically generated resources. To do so we should try to capture all the steps of the experiments and manual curation into an automatic pipeline. However, I do realize this is a great challenge. This approach would need to include the use of lab robotics, extensive vertical data integration and implementation of the dry-lab cycle principles as proposed in my thesis. Were we to succeed, then we should generate for each element, reproducible evidence. In this way we would be able to detect and remove errors from these important resources. However, this effort should be executed in collaboration with the data owners of these resources, otherwise it will result in yet another database, which is not synchronized to the original source.

Dry-lab cycle

I strongly believe that the dry-lab cycle previously introduced represents a powerful approach to extract information and knowledge from existing data. The information to knowledge step is a critical one, and problems associated with vertical data integration and provenance tracking need to be overcome. Therefore, within in my thesis, I have worked on developing data integration solutions needed to enable this dry-lab cycle. For some cases, I was able to perform some steps of this cycle such as the automatic creation of a genome annotation which includes the associated historical as well as the contextual provenance. However, I did not yet complete the cycle. In the following I would like to discuss an example on how this cycle could look like.

GEM are efficient tools to predict metabolic phenotypes. The following example shows how the dry-lab cycle can improve automatic GEM reconstruction. Figure 11.3 gives an overview of an example which includes a complete dry-lab cycle. The starting point is a genome sequence and known metabolic phenotypes. (i)In the domain discovery step all (bacterial) genomes are collected and *de-novo* automatically annotated with a gene prediction tool. A protocol such as the one given in Jérome Gouzy *et. al.* [204] can be used to search for new protein domains within the set of all available genomes. Subsequently

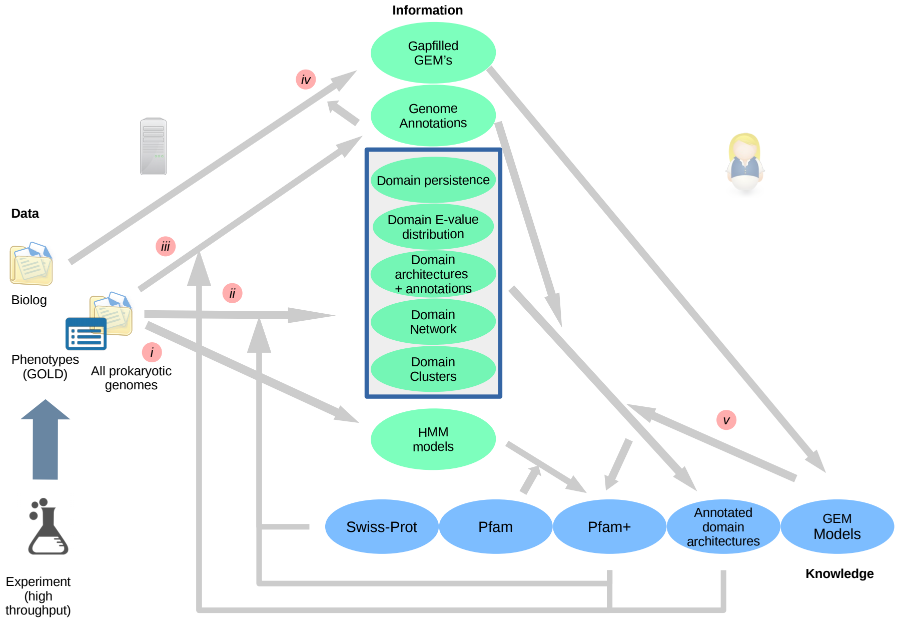


FIGURE 11.3: **An example of a dry-lab cycle.** The following steps are indicated: (i) Domain discovery from all (bacterial) genomes. (ii) Domain annotation and analysis. (iii) Genome annotation. (iv) GEM building. (v) The feedback step to improve the knowledge about the functions of the protein domains and domain architectures.

these domains are matched to already known domains in the Pfam database as, for example, done in Corin Yeats *et. al.* [533]. (ii) In the domain annotation step, an extensive analysis of the domain landscape is performed, on which domain persistence, domain E-value score distribution, domain architectures, domain co-occurrence networks and domain co-occurrence clusters are identified for each domain. Two domains are linked in the co-occurrence network if they both occur in the same protein. A set of domains form a cluster, if all members occur together in the same gene. Additionally, domains and domain architectures are associated to functional annotation based on the knowledge stored in Swiss-Prot database using, for instance, the method by Nam-Ninh Nguyen *et. al.* [354]. (iii) In the genome annotation step, the curated knowledge about domains and domain architectures is used to functionally annotate the target genome sequence. (iv) In the GEM building step, the functionally annotated genome is used to build a draft GEM. Gap-filling algorithms, like the one in **chapter 10** can then be used to integrate phenotype data (for example the Biolog Microarray Phenotype data) and produce a more accurate draft model. The GEM can be further curated and stored in a GEM database. (v) In the feedback step, the insights gained in the GEM building and gap filling process can be used again to improve the knowledge about the functions of the protein domain and domain architectures. For example, lets consider a GEM that contains a pathway with three reactions A, B and C. Let genes 'GenA' and 'GenC' be associated with reactions A and C respectively. Let's suppose that B has been added to the model as a result of gap filling, thus no gene can be associated to this reaction. Let 'GenB' within the same operon as 'GenA' and 'GenC'. Let 'GenB' have a single domain 'X' that is known to perform reaction D. Let this reaction D be equal to reaction B except its substrate and product molecule has one methyl group less. Then we can suggest that domain 'X' can also perform reaction B and will modify the annotation of 'GenB'.

This represents a relatively simple example of a dry-lab cycle. Additional steps could be included to further expand the cycle. Each added step requires additional data integration and keeping track of the associated provenance. I think it can result in solutions and pipelines integrating multiple methods in an iterative manner. Each iteration or cycle enhances the findings of other tools and results in knowledge with strong supporting evidence provenance track. I think this kind of solutions can lead to predictions that go beyond the possibilities of current methods and tools that use a limited set of sources with a limited use of the associated provenance. Every prediction needs experimental validation, however if we can use better prediction to generate experiments with an increased likelihood that the hypotheses is valid we can reduce the total cost of the needed experimental validation.

The application of semantic web technologies

Creation of pre-integrative information sets that can be readily queried and integrated requires the use of concepts and terms that can be matched across resources. This matching, requires the use of ontologies, so that a concept is defined within the context of other concepts and a concept can ultimately be seen as the set of links relating it to other concepts. Each concept can be referred to by a (set of) word(s) in a human language. Each word that refers to the concept is a term associated to the concept. Another requirement to generate pre-integrative information sets is that data are stored in a universal representation that can be readily interlinked with other resources and data sets. This can be achieved by representing the data as a graph in which the nodes represent instances of concepts and the links represent properties that describe the instance from which they originate.

Semantic web technologies offer solutions for both requirements. Earlier solutions within the semantic web technologies focus on the first requirement regarding the use of matchable concepts. These solutions include the OWL standard and associated reasoners. The OWL standard can be used to create ontologies to define concepts and to link these to the associated terms. The reasoners can be used to find duplications and inconsistencies within the concept definitions. Solutions regarding data linking include SPARQL, ShEx and

SHACL standards. SPARQL can be used to query the data, and the ShEx and SHACL standards can be used to validate the instance data, which ensures that the structure of the data follows the rules defined within the ontology. An example of instance data validation would be to verify that a protein has one and only one amino acid sequence associated to it.

Unfortunately, the growth of semantic web technologies has lead to using the technologies developed to solve the first requirement (concept matching) to solve the second requirement (data linking). For example the FALDO ontology uses the OWL standard to define its schema. I believe this to be a suboptimal approach, because the technologies developed to solve the first problem are not sufficient and sometimes incompatible with the second. The Stardog implementation uses OWL definitions in a closed world assumption to validate instance data [388]. However, as stated in the introduction of this thesis, the semantics and associated mathematical proofs of the OWL standard are defined with an open world assumption and therefore do not necessarily hold in with a closed world assumption and this might lead to inconsistencies. For instance, within the FALDO ontology a cardinality constraint is defined that states that **ExactPosition** must have exactly one position value. One can understand what it means. However within the open world assumptions this constraint can not be validated. If no position is given, the open world assumption states: It is not defined, however it could still exist somewhere in the open world and hence no error is given. In case two positions are given, it will report a violation error.

This mismatch between design principles and effective usage is, in my opinion, also present in the GO and SO ontologies. These are good solutions for their respective fields of application, namely characterization of gene function and nucleotide sequences. These ontologies have been successfully applied in several databases, including UniProt, to unambiguously reference to concepts within the field of biology. However, these ontologies can not be used to store all the information of the objects themselves within an interlinkable and reusable semantic data graph. For example, SO can be used to indicate that a part (indicated with FALDO) of a nucleotide sequence corresponds to a silencing RNA. However SO can not be used to describe all the properties of the silencing RNA as it can not describe what are the targets of the silencing RNA. Likewise, the GO ontology is successfully used to annotated genes with a given biological or molecular function, however it can not be used to describe a biochemical reaction that is not pre-included in the GO ontology.

Empusa

While working on this thesis I applied RDF2Graph, the tool described in **chapter 5** to several existing resources. My colleagues and I realized that many of the existing resources lack the quality required for efficient reuse and integration of the stored information. Moreover, we often found mismatches between the actual and the intended data structure and the one described in the OWL definition. I also noticed that the design principles needed for the schema definition for data import are different from the design principles needed to create ontologies. Therefore, I developed Empusa, described in **chapter 8**. Empusa is a tool that reads an ontology definition and generates code that can be used for data generation. Using Empusa we succeeded in creating the GBOL stack (**chapter 6**) and the associated tool SAPP chapter 7.

Developing GBOL and SAPP was greatly facilitated by Empusa. Moreover through this process we were able to assess the usability of the tool. A relative large number of master students have been involved in either research projects requiring accessing data encoded in the GBOL ontology or have been involved in developing modules interacting with the GBOL ontology. In all these cases we found that the use of the API and the supporting tools greatly improved the learning process of these students and reduced the amount of time required for autonomous work.

Currently, annotations for nearly 100.000 bacterial genomes have been converted into the GBOL format. However, it is likely that the GBOL ontology will be further modified in the future as it is not yet a community standards. A mechanism to automatically update already generated resources upon a modification in the GBOL ontology is still missing.

According to our design principles (specifically modularity and readability), we separated the sub-ontologies (value set) definition from the definition of the classes that have properties associated to them. However, a value set can evolve into a full ontology, which subsequently can further evolve into an ontology with classes that have properties, which in turn can point to another sub-ontology (value set). In GBOL, a value set can be defined for nucleic acids. Initially, it could be that only the *adenine*, *cytosine*, *guanine*, *thiamine*, *uracil* and *inosine* are included. In a second stage, this ontology could be extended with alternative forms. However, inclusion of all alternative forms and modifications would cause the complexity of the ontology to explode. Thus, instead of adding values to the value set a class with properties describing the chemical representation could be added. In turn this new class with chemical representations could have a property linking to an another value set describing the type of chemical links. This constant growth of sub-ontologies into ontologies is to be expected, as tools and ontologies are further used. Currently, no solution exists to automatically adapt to this growth of ontologies. In the future, I would like to add such a functionality to Empusa trough the use of meta-class definition schemas.

Genome annotation & automated draft GEM reconstruction

Thiele et. al. [494] describe a protocol to create a high quality, manually curated GEM. The approach I have presented here relates to the automatic reconstruction of (draft) GEMs. This method depends on reference databases of known metabolic pathways and it only partly overlaps with the protocol by

[494]. Automatic draft GEM reconstruction based on reference databases entails, among possible other, four important steps: i) Gene identification in the genome sequence; (ii) gene functional annotation; (iii) linkage of of functional annotation and reactions within the selected database; (iv) model completion using gap-filling methods. In the following, I will discuss how these stpes could be further improved using pre-integrative data sets and principles of the dry-lab cycle.

Gene identification

Multiple gene identification tools exist. Currently, Prodigal [235] is commonly applied for prokaryotic genomes. This method uses a single genome to train itself in three phases using a dynamic programming approach. The first phase is based on the GC bias per codon position (1, 2 or 3) and the result is optimized so that predicted genes have high GC score for the selected codon position. The second phase is based on hexamers, each hexamer has a within gene occurrence and an outside gene occurrence score. The result is optimized so that genes contain hexamers that are most likely to occur within a gene. The third phase is based on the ribosomal binding sites (RBS). The result is optimized so that predicted genes have a strong RBS signal in which the RBS binding motif is modified throughout the cycles. The combination of these methods leads to accurate gene prediction. However, the approach might be further enhanced by including pre-integrative information from other available bacterial genomes and genes. In this way, information from closely related species could be included, thus enabling the training on multiple genomes within the same genus. This information can be used to determine homologs within the same clade, which in turn can be used to calculate specific codon usage bias, and nucleotide versus amino acid mutation rates. This information could be used to identify differences between the positions upstream and downstream of the start codon, ultimately leading to better gene model predictions.

Gene function prediction

Traditional methods of computational gene functional annotation are based on sequence similarity identified using algorithms such BLAST [15], wheras the HMMER rely on protein domains for the annotation [152]. Several databases exist containing motifs to recognize these domains and many have been integrated into the InterPro database [170].

I believe that functional annotation based on domains could be improved. (i) Currently, domains are manually added to InterPro and domain models are mostly based on hidden Markov models (HMM). Model incorporation into the databases could be further expanded by automatically mining all available genome sequences to identify new domains, specially in proteins annotated as hypothetical proteins. For the newly found domains, provenance should be available and linked to the existing manually curated elements, as discussed above (see step i in Figure 11.3). Trough the use of domain co-occurrence and domain clusters (see step ii in Figure 11.3) and gene networks as presented in **chapter 2** the newly identified domains can be functionally annotated. (ii) MRFalign [310] can, in contrast to HMM, capture correlation between sites at longer distances in the predictions model and would therefore further improve the sensitivity. For example, if the amino acid at position 10 of a given protein is an alanine, then at position 30 a tyrosine is to be expected. (iii) The distribution of the E-values of all hits associated to a domain should be systematically explored, as it can lead to more specific domain definition, as illustrated in Figure 11.2.

Within bacterial genomes some domains are often found in in commonly occurring combinations, called domain architectures. A complete collection of domain architectures is available and browseable at the EBI InterPro website [170]. EnzDP uses domain architectures to functionally annotate genomes [354]. Domain architectures enable the identification of domain co-occurences in proteins. Domain co-occurence can be used to built networks and within these networks, sub-networks or clusters do appear that represent highly modular systems. For example the cluster of domains related to the synthesis of complex polyketides (Figure 11.4). Unique combinations of these domains form large (>10 domains within one protein) proteins that synthesize highly specialized molecules with a polykytide backbone. This knowledge has been used to create specific annotation tools such as Anti-Smash [522]. Identification of clusters of commonly co-occurring domains can further refine the approach. We have noticed that within these clusters it is often found some domains forming a 'core' whereas other are mutually exclusive. Each of these mutually exclusive members tend to give a unique function to the protein containing the cluster of domains. However, further experimental information is still needed to prove this.

Draft GEM building, linking annotation to chemical reactions

GEM reconstruction requires mapping functional annotations to chemical reactions within the selected name space, which could be Model Seed [137] or MetaCyc [95] as described in **chapter 10** [494]. However, in many cases direct mappings to these name spaces are missing. Instead, the mappings are dependent on EC numbers, GO terms or an identifier of another name space. These can be further mapped to the selected name space. However, this task is hampered by the use of different metabolite and reaction identifiers and description between the databases, the so called name space problem. Technical translation issues between databases could be solved using InChI strings, that can uniquely identify compounds (and conformations) and can be used to uniquely identify reactions. Thus, a reaction would be represented as a unique transformation between two sets of compounds, products and substrates.

However, the matching between genes and reactions is often hampered by the lack of complete information on preferred substrate specificity and conformation or cofactor utilization. These could be solved by the use of generic

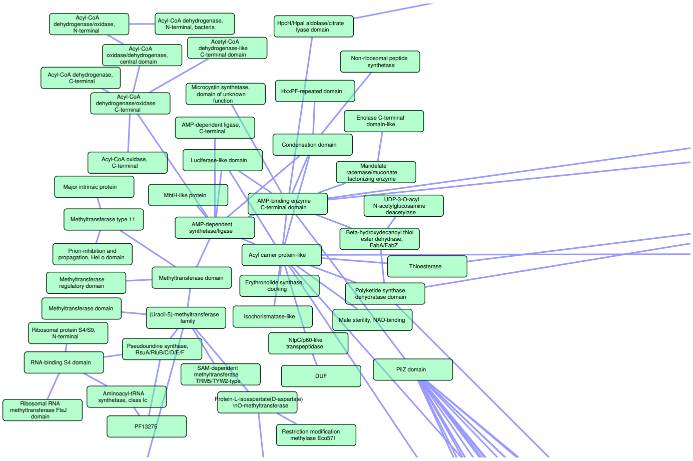

Acyl-CoA dehydrogenase/oxidase, N-terminal
Acyl-CoA dehydrogenase, N-terminal, bacteria
HpcH/HpaI aldolase/citrate lyase domain
Acyl-CoA oxidase/dehydrogenase, central domain
Acetyl-CoA dehydrogenase-like C-terminal domain
Non-ribosomal peptide synthetase
Acyl-CoA dehydrogenase, C-terminal
Microcystin synthetase, domain of unknown function
HxxPF-repeated domain
Acyl-CoA dehydrogenase/oxidase C-terminal
Enolase C-terminal domain-like
AMP-dependent ligase, C-terminal
Condensation domain
Acyl-CoA oxidase, C-terminal
Luciferase-like domain
Mandelate racemase/muconate lactonizing enzyme
Major intrinsic protein
MbtH-like protein
AMP-binding enzyme C-terminal domain
UDP-3-O-acyl N-acetylglucosamine deacetylase
Methyltransferase type 11
AMP-dependent synthetase/ligase
Beta-hydroxydecanoyl thiol ester dehydrase, FabA/FabZ
Prion-inhibition and propagation, HeLo domain
Acyl carrier protein-like
Thioesterase
Methyltransferase regulatory domain
Methyltransferase domain
Erythronolide synthase, docking
Polyketide synthase, dehydratase domain
Methyltransferase domain
(Uracil-5)-methyltransferase family
Isochorismatase-like
Male sterility, NAD-binding
Ribosomal protein S4/S9, N-terminal
NlpC/p60-like transpeptidase
Pseudouridine synthase, RsuA/RluB/C/D/E/F
RNA-binding S4 domain
SAM-dependent methyltransferase TRM5/TYW2-type
DUF
PilZ domain
Aminoacyl-tRNA synthetase, class Ic
Protein-L-isoaspartate(D-aspartate) \nO-methyltransferase
Ribosomal RNA methyltransferase FtsJ domain
PF13275
Restriction modification methylase Eco57I

FIGURE 11.4: **Domain cluster related to polyketide synthesis (PKS).** A cluster within a domain co-occurrence network. In this network two domains appear linked if the co-occur in the same protein in any of the studied genomes. The represented cluster contains domains related to the synthesis of complex molecules with a polyketide backbone.

compound classes. A generic class could be defined to represent all known subclasses and conformations of the substrate a given enzyme can accept. In the same way it is often unclear what is the preferred cofactor of an enzyme in a given organism. A solution would be to use a generic compound class for the cofactor.

Therefore, compound classes are a good solution to solve these problems, however the use of compound classes is associated with other problems: (i)Existing reference databases (without generic compounds) contain entries that defines a specific compound or cofactor for which there is no supporting evidence or the available evidence only supports general compound class. For example a reaction is defined to use NADH, while it is only proven that it uses either NADH or NADPH. (ii) For GEM building it is important to ensure that this generic compound classes are instantiated to specific compounds so that it does not allow for unwanted inter-conversions and unbalances in the model.

Overcoming these issues requires detailed provenance information on the gene-protein-reaction (GPR) associations, which includes for which substrate specificity there is supporting evidence. In this way, the process of model manual curation can consider the supporting evidence for each reaction. The provenance information should be included in the GEM and should contain: the functional gene annotation, from gene to reaction annotation, general compound to specific substrate instantiation and any subsequent modifications in the GEM.

It should be noted that still many challenges would need to be addressed to improve the GPR associations, such as the associations for moonlighting enzymes, enzymes with broad range of substrates, and differences due to changes in conditions such as pH, salinity or temperature, among other.

GEM building, gap filling

As introduced in **chapter 10**, a number of tools exist to integrate experimental data (growth phenotype data, knockout data and C13 flux data) with GEMs to increase the quality of the reconstructions [545]. For most of these tools no ready to use implementations nor standardization of the needed input data are available. I think the community would greatly benefit from efforts to make these tools available and of releasing the relevant experimental data as FAIR data.

Moreover, as indicated in the previous section the translation from the functional annotation to a reaction list would return a set of reactions containing generic reactions. These generic reactions can not be used in the FBA analysis. Special methods would need to be developed to select the specific reactions and cofactors needed in the context of the model. It is important that these selected specific reactions are marked so that a curator would know that there is no supporting experimental information available.

The resulting gap filled GEMs together with the provenance data of the annotation and gap filling processes can be used to identify new enzymes and pathways. Orphan reactions in a GEM resulting from gap-filling provide important information on enzymes not yet fully characterized. This information can be used to further mine the genome sequence and look up other related reactions. Guilt-by-association principles can be used to pinpoint the responsible enzyme. Most often it would be linked to incomplete functional annotations, due mainly to threshold settings too stringent for the particular enzymes, but it also holds the potential to pinpoint novel enzymes. Similarly, novel pathways can be assembled if a chain of general reactions can be identified. This can lead to the identification of species specific pathways, such as the cholesterol degradation pathway in *M. tuberculosis* [412].

GEM re-use

Extension of published GEMs is hampered by the lack of provenance details on the manual curation. To further expand and curate a GEM one would need to re-curate all existing reactions. For optimal re-usability, GEMs should incorporate provenance on the functional annotations, the linking to reactions within the selected name space and the automatic gap filling. A clear distinction should be visible between the automatically generated information and that created with manual curation. Moreover, it should be possible to rerun methods used for the automatically generated parts, without losing the manual curation work, except for the conflicting elements.

Model information should be encoded in the RDF data model, otherwise I think it would be too complex to encode all the provenance information. An exporter and importer to the current community standard (SBML) would be required. SBML has the support to include, for each element. a piece of RDF data, so it is possible to export all information into the SBML format. For optimal reusability at least one default biomass definition and one minimal media should be co-encoded into the GEM. Finally, the GEM file should be stored in a git versioning system so that the complete history of the GEM is available.

From Genotype to Phenotype

GEMs can be used to predict metabolic phenotypes, however they have limited use to predict other physiological properties such as pathogenicity or antibiotic resistance. Machine learning based methods that use various genomic properties can be used to predict these phenotypes [23, 133].

Machine learning methods require training data sets. Creation of these training sets requires standardized and normalized annotations and standardized description of phenotypes. The GBOL (**chapter 6**) ontology in combination with the SAPP annotation pipelines (**chapter 7**) ensures that the genetic information is readily available in a standardized and normalized format. The phenotype description can be standardized using a bacterial phenotype ontology [106]. These approaches could also be used to identify the genetic determinants of a given phenotype.

Bibliography

[1] Abdallah, A. M. et al. “Type VII secretion–mycobacteria show the way”. In: *Nature Reviews Microbiology* (2007).

[2] Abramovitch, R. B. et al. “aprABC: a *Mycobacterium tuberculosis* complex-specific locus that modulates pH-driven adaptation to the macrophage phagosome”. In: *Molecular Microbiology* (2011).

[3] Acevedo-Rocha, C. G. et al. “From essential to persistent genes: a functional approach to constructing synthetic life”. In: *Trends in Genetics* (2013).

[4] Agarwal, N., Raghunand, T. R., and Bishai, W. R. “Regulation of the expression of whiB1 in *Mycobacterium tuberculosis*: role of cAMP receptor protein”. In: *Microbiology* (2006).

[5] Agarwal, N. et al. “Cyclic AMP intoxication of macrophages by a *Mycobacterium tuberculosis* adenylate cyclase”. In: *Nature* (2009).

[6] Agranoff, D et al. “*Mycobacterium tuberculosis* expresses a novel pH-dependent divalent cation transporter belongiang to the Nramp family”. In: *The Journal of Experimental Medicine* (1999).

[7] Ahn, S. H. et al. “Gene Expression-Based Classifiers Identify *Staphylococcus* aureus Infection in Mice and Humans”. In: *PLoS ONE* (2013).

[8] Akhter, Y. et al. “Genome scale portrait of cAMP-receptor protein (CRP) regulons in mycobacteria points to their role in pathogenesis”. In: *Gene* (2008).

[9] Alako, B. T. F. et al. “TreeDomViewer: a tool for the visualization of phylogeny and protein domain structure”. In: *Nucleic Acids Research* (2006).

[10] Alam, M. S., Garg, S. K., and Agrawal, P. “Studies on structural and functional divergence among seven WhiB proteins of *Mycobacterium tuberculosis H37Rv*”. In: *FEBS Journal* (2009).

[11] Alcántara, R. et al. “Rhea - A manually curated resource of biochemical reactions”. In: *Nucleic Acids Research* (2012).

[12] Alföldi, J. and Lindblad-Toh, K. “Comparative genomics as a tool to understand evolution and disease”. In: *Genome Research* (2013).

[13] Altay, G and Emmert-Streib, F. “Inferring the conservative causal core of gene regulatory networks”. In: *BMC Systems Biology* (2010).

[14] Altay, G et al. “Differential C3NET reveals disease networks of direct physical interactions”. In: *BMC Bioinformatics* (2011).

[15] Altschul, S. F. et al. “Basic local alignment search tool”. In: *Journal of Molecular Biology* (1990).

[16] Amara, U. et al. “Interaction Between the Coagulation and Complement System”. In: *Advances in experimental medicine and biology*. 2008.

[17] Antezana, E., Mironov, V., and Kuiper, M. “The emergence of Semantic Systems Biology”. In: *New Biotechnology* (2013).

[18] Antoniou, G. and Van Harmelen, F. “Web ontology language: Owl”. In: *Handbook on ontologies*. 2004.

[19] Antunes, I. and Kassiotis, G. "Suppression of Innate Immune Pathology by Regulatory T Cells during Influenza A Virus Infection of Immunodeficient Mice". In: *Journal of Virology* (2010).

[20] Apache Foundation. *Maven*. 2001. URL: `http://maven.apache.org/index.html`.

[21] Apache Foundation. *Apache Jena*. 2013. URL: `http://jena.apache.org/`.

[22] Aranda, C. B. et al. *SPARQL 1.1 Overview*. 2013. URL: `https://www.w3.org/TR/sparql11-overview/`.

[23] Arango-Argoty, G. et al. "DeepARG: a deep learning approach for predicting antibiotic resistance genes from metagenomic data". In: *Microbiome* (2018).

[24] Ark, K. C. van der et al. "More than just a gut feeling: constraint-based genome-scale metabolic models for predicting functions of human intestinal microbes". In: *Microbiome* (2017).

[25] Ark, K. C. van der et al. "Model-driven design of a minimal medium for Akkermansia muciniphila confirms mucus adaptation". In: *Microbial Biotechnology* (2018).

[26] Arya, S. et al. "Truncated hemoglobin, HbN, is post-translationally modified in *Mycobacterium tuberculosis* and modulates host-pathogen interactions during intracellular infection". In: *Journal of Biological Chemistry* (2013).

[27] Ascenzi, P. et al. "Isoniazid inhibits the heme-based reactivity of *Mycobacterium tuberculosis* truncated hemoglobin N". In: *PLoS ONE* (2013).

[28] Ashburner, M et al. "Gene Ontology: Tool for The Unification of Biology". In: *Nature Genetics* (2000).

[29] Ates, L. S. and Brosch, R. "Discovery of the type VII ESX-1 secretion needle?" In: *Molecular Microbiology* (2017).

[30] Augenstreich, J. et al. "ESX - 1 and phthiocerol dimycocerosates of *Mycobacterium tuberculosis* act in concert to cause phagosomal rupture and host cell apoptosis". In: *Cellular Microbiology* (2017).

[31] Aziz, R. K. et al. "The RAST Server: rapid annotations using subsystems technology". In: *BMC Genomics* (2008).

[32] Bai, G., Knapp, G. S., and McDonough, K. A. "Cyclic AMP signalling in mycobacteria: redirecting the conversation with a common currency". In: *Cellular Microbiology* (2011).

[33] Bai, G. et al. "Characterization of *Mycobacterium tuberculosis* Rv3676 (CRP Mt), a Cyclic AMP Receptor Protein-Like DNA Binding Protein". In: *Journal of Bacteriology* (2005).

[34] Bailey, T. L. and Elkan, C. "Fitting a Mixture Model by Expectation Maximization to Discover Motifs in Biopolymers". In: *Proceedings of the Second International Conference on Intelligent Systems for Molecular Biology* (1994).

[35] Bairoch, A and Apweiler, R. "The SWISS-PROT protein sequence database and its supplement TrEMBL in 2000". In: *Nucleic Acids Research* (2000).

[36] Balázsi, G et al. "The temporal response of the *Mycobacterium tuberculosis* gene regulatory network during growth arrest". In: *Molecular Systems Biology* (2008).

[37] Baltrus, D. a. et al. "Dynamic evolution of pathogenicity revealed by sequencing and comparative genomics of 19 *Pseudomonas syringae* isolates". In: *PLoS Pathogens* (2011).

[38] Banchereau, R. et al. "Host immune transcriptional profiles reflect the variability in clinical disease manifestations in patients with *staphylococcus aureus* infections". In: *PLoS ONE* (2012).

[39] Banerjee, S. et al. "Iron-dependent RNA-binding activity of *Mycobacterium tuberculosis* aconitase". In: *Journal of Bacteriology* (2007).

[40] Bansal, M et al. "How to infer gene networks from expression profiles". In: *Molecular Systems Biology* (2007).

[41] Bansal, R. and Kumar, V. A. "*Mycobacterium tuberculosis* virulence-regulator PhoP interacts with alternative sigma factor SigE during acid-stress response". In: *Molecular Microbiology* (2017).

[42] Bao, J. et al. *OWL 2 Web Ontology Language Document Overview (Second Edition)*. 2012. URL: https://www.w3.org/TR/owl2-overview/.

[43] Bard, J. B. L. and Rhee, S. Y. "Ontologies in biology: design, applications and future challenges". In: *Nature Reviews Genetics* (2004).

[44] Barik, S. et al. "RseA, the SigE specific anti-sigma factor of *Mycobacterium tuberculosis*, is inactivated by phosphorylation-dependent ClpC1P2 proteolysis". In: *Molecular Microbiology* (2010).

[45] Barrett, T et al. "NCBI GEO: archive for functional genomics data sets –10years on". In: *Nucleic Acids Research* (2010).

[46] Bartek, I. L. et al. "The DosR regulon of *M. tuberculosis* and antibacterial tolerance". In: *Tuberculosis* (2009).

[47] Barua, D., Kim, J., and Reed, J. L. "An automated phenotype-driven approach (GeneForce) for refining metabolic and regulatory models". In: *PLoS Computational Biology* (2010).

[48] Basso, K. et al. "Reverse engineering of regulatory networks in human B cells". In: *Nature Genetics* (2005).

[49] Batagelj, V. and Mrvar, A. "Pajek — Analysis and Visualization of Large Networks". In: *Graph Drawing Software*. 2004.

[50] Bateman, A. et al. "UniProt: The universal protein knowledgebase". In: *Nucleic Acids Research* (2017).

[51] Beaulieu, A. *Learning SQL*. 2009.

[52] Beckett, D. and Berners-Lee, T. *Turtle - Terse RDF Triple Language*. 2011. URL: https://www.w3.org/TeamSubmission/turtle/.

[53] Belcastro, V et al. "Transcriptional gene network inference from a massive dataset elucidates transcriptome organization and gene function". In: *Nucleic Acids Research* (2011).

[54] Belda, E. et al. "The revisited genome of Pseudomonas putida KT2440 enlightens its value as a robust metabolic chassis". In: *Environmental Microbiology* (2016).

[55] Belleau, F. et al. "Bio2RDF: towards a mashup to build bioinformatics knowledge systems". In: *Journal of Biomedical Informatics* (2008).

[56] Ben-Kiki, O., Evans, C., and Net, I. dot. *YAML Ain't Markup Language (YAML) Version 1.2*. 2009. URL: http://www.yaml.org/spec/1.2/spec.html.

[57] Benson, D. A. et al. "GenBank". In: *Nucleic Acids Research* (2013).

[58] Berners-Lee, T., Hendler, J., and Lassila, O. "The Semantic Web. A new form of Web content that is meaningful to computers will unleash a revolution of new possibilities". In: *Scientific American* (2001).

[59] Bernini, P. et al. "The cardiovascular risk of healthy individuals studied by NMR metabonomics of plasma samples". In: *Journal of Proteome Research* (2011).

[60] Bertels, F. et al. "Automated Reconstruction of Whole-Genome Phylogenies from Short-Sequence Reads". In: *Molecular Biology and Evolution* (2014).

[61] Bitter, W. et al. "Systematic genetic nomenclature for type VII secretion systems". In: *PLoS Pathogens* (2009).

[62] Bixby, R. E. "Solving Real-World Linear Programs: A Decade and More of Progress". In: *Operations Research* (2002).

[63] Bizer, C., Heath, T., and Berners-Lee, T. "Linked data-the story so far". In: *Semantic Services, Interoperability and Web Applications: Emerging Concepts* (2009).

[64] Blanchette, C. D. et al. “Decoupling internalization, acidification and phagosomal-endosomal/lysosomal fusion during phagocytosis of InlA coated beads in epithelial cells”. In: *PLoS ONE* (2009).

[65] Blankenberg, D. et al. “Galaxy: A web-based genome analysis tool for experimentalists”. In: *Current Protocols in Molecular Biology* (2010).

[66] Blasco, B. et al. “Virulence regulator EspR of *Mycobacterium tuberculosis* is a nucleoid-associated protein”. In: *PLoS Pathogens* (2012).

[67] Bochner, B. R., Gadzinski, P, and Panomitros, E. “Phenotype Microarrays for high-throughput phenotypic testing and assay of gene function”. In: *Genome Research* (2001).

[68] Bolleman, J. T. et al. “FALDO: A semantic standard for describing the location of nucleotide and protein feature annotation”. In: *Journal of Biomedical Semantics* (2016).

[69] Boneva, I. et al. “Validating RDF with Shape Expressions”. In: *arXiv:1404.1270* (2014).

[70] Bonneau, R et al. “The Inferelator: an algorithm for learning parsimonious regulatory networks from systems-biology data sets de novo”. In: *Genome Biology* (2006).

[71] Borgos, S. E. F. et al. “Mapping global effects of the anti-sigma factor MucA in Pseudomonas fluorescens SBW25 through genome-scale metabolic modeling”. In: *BMC Systems Biology* (2013).

[72] Boshoff, H. I. M. et al. “The Transcriptional Responses of *Mycobacterium tuberculosis* to Inhibitors of Metabolism”. In: *Journal of Biological Chemistry* (2004).

[73] Bowers, P. M. et al. “Prolinks: a database of protein functional linkages derived from coevolution”. In: *Genome Biology* (2004).

[74] Brachman, R. J. *A Structural Paradigm for Representing Knowledge*. 1977.

[75] Brady, R. A., Bruno, V. M., and Burns, D. L. “RNA-seq analysis of the host response to *Staphylococcus aureus* skin and soft tissue infection in a mouse model”. In: *PLoS ONE* (2015).

[76] Bretl, D. J., Demetriadou, C., and Zahrt, T. C. “Adaptation to environmental stimuli within the host: two-component signal transduction systems of *Mycobacterium tuberculosis*”. In: *Microbiology and Molecular Biology Reviews* (2011).

[77] Bretl, D. J. et al. “MprA and DosR coregulate a *Mycobacterium tuberculosis* virulence operon encoding Rv1813c and Rv1812c”. In: *Infection and Immunity* (2012).

[78] Bretl, D. J. et al. “The MprB extracytoplasmic domain negatively regulates activation of the *Mycobacterium tuberculosis MprAB* two-component system”. In: *Journal of Bacteriology* (2014).

[79] Brickley, D. and Guha, R. V. *RDF Vocabulary Description Language 1.0: RDF Schema*. 2004. URL: `http://www.w3.org/TR/2004/REC-rdf-schema-20040210/`.

[80] Brickley, D., Guha, R. V., and McBride, B. *RDF Schema 1.1*. 2014. URL: `http://www.w3.org/TR/rdf-schema/`.

[81] Brickley, D. and Miller, L. *FOAF vocabulary specification 0.91*. 2007. URL: `http://xmlns.com/foaf/spec/20071002.html`.

[82] Brown, K. R. et al. “NAViGaTOR: Network analysis, visualization and graphing Toronto”. In: *Bioinformatics* (2009).

[83] Brown, M. R. W. and Kornberg, A. “Inorganic polyphosphate in the origin and survival of species”. In: *PNAS* (2004).

[84] Bruggeman, F. J. et al. “Introduction to systems biology”. In: *Plant Systems Biology*. 2007.

[85] Buchmeier, N et al. “A parallel intraphagosomal survival strategy shared by *Mycobacterium tuberculosis* and *Salmonella enterica*”. In: *Molecular Microbiology* (2000).

[86] Bunker, R. D. et al. “A functional role of Rv1738 in *Mycobacterium tuberculosis* persistence suggested by racemic protein crystallography”. In: *Proceedings of the National Academy of Sciences of the United States of America of the United States of America* (2015).

[87] Burgard, A. P. and Maranas, C. D. “Probing the performance limits of the *Escherichia coli* metabolic network subject to gene additions or deletions”. In: *Biotechnology and Bioengineering* (2001).

[88] Butala, M, Žgur-Bertok, D, and Busby, S. J. W. “The bacterial LexA transcriptional repressor”. In: *Cell Molecular Life Sciences* (2008).

[89] Butler, D. “Scientists: Your number is up”. In: *Nature* (2012).

[90] Camacho, C et al. “BLAST+: architecture and applications”. In: *BMC Bioinformatics* (2009).

[91] Cantone, I et al. “A yeast synthetic network for in vivo assessment of reverse-engineering and modeling approaches”. In: *Cell* (2009).

[92] Carbon, S. et al. “Expansion of the gene ontology knowledgebase and resources: The gene ontology consortium”. In: *Nucleic Acids Research* (2017).

[93] Caricilli, A. M. and Saad, M. J. A. “The role of gut microbiota on insulin resistance”. In: *Nutrients* (2013).

[94] Casonato, S. et al. “WhiB5, a transcriptional regulator that contributes to *Mycobacterium tuberculosis* virulence and reactivation”. In: *Infection and Immunity* (2012).

[95] Caspi, R. et al. “The *MetaCyc* database of metabolic pathways and enzymes and the *BioCyc* collection of pathway/genome databases”. In: *Nucleic Acids Research* (2012).

[96] Cerans, K. et al. “Advanced OWL 2.0 Ontology Visualization in OWLGrEd”. In: *Databases and Information Systems VII* (2012).

[97] Champion, O. L. et al. “*Yersinia pseudotuberculosis* mntH functions in intracellular manganese accumulation, which is essential for virulence and survival in cells expressing functional Nramp1”. In: *Microbiology* (2011).

[98] Chao, M. C. and Rubin, E. J. “Letting Sleeping dos Lie: Does Dormancy Play a Role in Tuberculosis?” In: *Annual Review of Microbiology* (2010).

[99] Chauhan, A et al. “Interference of *Mycobacterium tuberculosis* cell division by Rv2719c, a cell wall hydrolase”. In: *Molecular Microbiology* (2006).

[100] Chauhan, S. and Tyagi, J. S. “Cooperative binding of phosphorylated DevR to upstream sites is necessary and sufficient for activation of the Rv3134c-devRS operon in *Mycobacterium tuberculosis*: implication in the induction of DevR target genes”. In: *Journal of Bacteriology* (2008).

[101] Chauhan, S. et al. “Comprehensive insights into *Mycobacterium tuberculosis* DevR (DosR) regulon activation switch”. In: *Nucleic Acids Research* (2011).

[102] Chawla, M. et al. “*Mycobacterium tuberculosis* WhiB4 regulates oxidative stress response to modulate survival and dissemination in vivo”. In: *Molecular Microbiology* (2012).

[103] Chen, H and Boutros, P. C. “VennDiagram: a package for the generation of highly-customizable Venn and Euler diagrams in R”. In: *BMC Bioinformatics* (2011).

[104] Chen, J. M. et al. “EspD is critical for the virulence-mediating ESX-1 secretion system in *Mycobacterium tuberculosis*”. In: *Journal of Bacteriology* (2012).

[105] Chen, Z. et al. "Mycobacterial *WhiB6* Differentially Regulates ESX-1 and the Dos Regulon to Modulate Granuloma Formation and Virulence in Zebrafish". In: *Cell Reports* (2016).

[106] Chibucos, M. C. et al. "An ontology for microbial phenotypes". In: *BMC Microbiology* (2014).

[107] Chibucos, M. C. et al. "Standardized description of scientific evidence using the Evidence Ontology (ECO)". In: *Database* (2014).

[108] Cho, D.-Y., Kim, Y.-A., and Przytycka, T. M. "Chapter 5: Network Biology Approach to Complex Diseases". In: *PLoS Computational Biology* (2012).

[109] Chuang, H.-Y., Hofree, M., and Ideker, T. "A Decade of Systems Biology". In: *Annual Review of Cell and Developmental Biology* (2010).

[110] Chuang, Y.-m. et al. "Stringent Response Factors PPX1 and PPK2 Play an Important Role in *Mycobacterium tuberculosis* Metabolism, Biofilm Formation, and Sensitivity to Isoniazid In Vivo". In: *Antimicrobial Agents and Chemotherapy* (2016).

[111] Clemmensen, H. S. et al. "An attenuated *Mycobacterium tuberculosis* clinical strain with a defect in ESX-1 secretion induces minimal host immune responses and pathology". In: *Scientific Reports* (2017).

[112] Cochrane, G. et al. "Facing growth in the European Nucleotide Archive". In: *Nucleic Acids Research* (2013).

[113] Colangeli, R et al. "The multifunctional histone-like protein Lsr2 protects mycobacteria against reactive oxygen intermediates". In: *PNAS* (2009).

[114] Colangeli, R. et al. "Transcriptional regulation of multi-drug tolerance and antibiotic-induced responses by the histone-like protein Lsr2 in *M. tuberculosis*". In: *PLoS Pathogens* (2007).

[115] Conrad, W. H. et al. "Mycobacterial ESX-1 secretion system mediates host cell lysis through bacterium contact-dependent gross membrane disruptions". In: *Proceedings of the National Academy of Sciences of the United States of America of the United States of America* (2017).

[116] Cook, H. and Ussery, D. W. "Sigma factors in a thousand *E. coli* genomes". In: *Environmental Microbiology* (2013).

[117] Cook, J. et al. "Consensus on consensus: A synthesis of consensus estimates on human-caused global warming". In: *Environmental Research Letters* (2016).

[118] Cooper, D. N. et al. "Where genotype is not predictive of phenotype: towards an understanding of the molecular basis of reduced penetrance in human inherited disease". In: *Human Genetics* (2013).

[119] Croft, D. et al. "The Reactome pathway knowledgebase". In: *Nucleic Acids Research* (2014).

[120] Csardi, G. and Nepusz, T. "The igraph software package for complex network research". In: *International Journal of Neural Systems* (2006).

[121] Cyktor, J. C. et al. "IL-10 inhibits mature fibrotic granuloma formation during *Mycobacterium tuberculosis* infection". In: *Journal of Immunology* (2013).

[122] Daim, S. et al. "Expression of the *Mycobacterium tuberculosis* PPE37 protein in *Mycobacterium smegmatis* induces low tumour necrosis factor alpha and interleukin 6 production in murine macrophages". In: *Journal of Medical Microbiology* (2011).

[123] Dam, J. C. J. van et al. "Integration of heterogeneous molecular networks to unravel gene-regulation in *Mycobacterium tuberculosis*". In: *BMC Medical Genomics* (2014).

[124] Dam, J. C. J. van et al. "Interoperable genome annotation with *GBOL*, an extendable infrastructure for functional data mining". In: *bioRxiv* (2017).

[125] Dam, J. C. van et al. "RDF2Graph a tool to recover, understand and validate the ontology of an RDF resource". In: *Journal of Biomedical Semantics* (2015).

[126] Daniel, J. et al. "Rv2477c is an antibiotic-sensitive manganese-dependent ABC-F ATPase in *Mycobacterium tuberculosis*". In: *Biochemical and Biophysical Research Communications* (2017).

[127] Dass, B. K. M. et al. "Cyclic AMP in mycobacteria: characterization and functional role of the Rv1647 ortholog in *Mycobacterium smegmatis*". In: *Journal of Bacteriology* (2008).

[128] Daub, C. O. et al. "Estimating mutual information using B-spline functions – an improved similarity measure for analysing gene expression data". In: *BMC Bioinformatics* (2004).

[129] Davis, E. O., Dullaghan, E. M., and Rand, L. "Definition of the Mycobacterial SOS Box and Use To Identify LexA-Regulated Genes in *Mycobacterium tuberculosis*". In: *Journal of Bacteriology* (2002).

[130] De Smet, R and Marchal, K. "Advantages and limitations of current network inference methods". In: *Nature Reviews Microbiology* (2010).

[131] Deb, C et al. "A Novel In Vitro Multiple-Stress Dormancy Model for *Mycobacterium tuberculosis* Generates a Lipid-Loaded, Drug-Tolerant, Dormant Pathogen". In: *PLoS ONE* (2009).

[132] Degtyarenko, K. et al. "ChEBI: A database and ontology for chemical entities of biological interest". In: *Nucleic Acids Research* (2008).

[133] Deneke, C., Rentzsch, R., and Renard, B. Y. "PaPrBaG: A machine learning approach for the detection of novel pathogens from NGS data". In: *Scientific Reports* (2017).

[134] Deng, J. *Gene Essentiality: Methods and Protocols*. 2015. Chap. An Integra.

[135] Deng, J. et al. "Investigating the predictability of essential genes across distantly related organisms using an integrative approach". In: *Nucleic Acids Research* (2011).

[136] Deng, W., Xiang, X., and Xie, J. "Comparative genomic and proteomic anatomy of Mycobacterium ubiquitous Esx family proteins: implications in pathogenicity and virulence". In: *Current Microbiology* (2014).

[137] Devoid, S. et al. "Automated genome annotation and metabolic model reconstruction in the SEED and Model SEED". In: *Methods in Molecular Biology* (2013).

[138] Dewitt, M. A. et al. "The Conformations of the Manganese Transport Regulator of Bacillus subtilis in its Metal-free State". In: *Journal of Molecular Biology* (2007).

[139] Diaz-ochoa, V. E. et al. "*Salmonella* Mitigates Oxidative Stress and Thrives in the Inflamed Gut by Evading Calprotectin-Mediated Manganese Sequestration". In: *Cell Host & Microbe* (2016).

[140] Dietzold, J., Gopalakrishnan, A., and Salgame, P. "Duality of lipid mediators in host response against *Mycobacterium tuberculosis*: good cop, bad cop". In: *F1000Prime Reports* (2015).

[141] *DigiSal, Towards the Digital Salmon: From a reactive to a pre-emptive research strategy in aquaculture*. 2017. URL: `https://www.forskningsradet.no/prosjektbanken/\#/project/NFR/248792/Sprak=en`.

[142] DiGiuseppe Champion, P. A. et al. "ESX-1 secreted virulence factors are recognized by multiple cytosolic AAA ATPases in mycobactria". In: *Molecular Microbiology* (2009).

[143] Dijkstra, E. W. "A note on two problems in connexion with graphs". In: *Numerische Mathematik* (1959).

[144] Dimmer, E. C. et al. "The UniProt-GO Annotation database in 2011". In: *Nucleic Acids Research* (2012).

[145] Dittrich, D. et al. “Characterization of a *Mycobacterium tuberculosis* mutant deficient in pH-sensing adenylate cyclase Rv1264”. In: *International Journal of Medical Microbiology* (2006).

[146] Doerks, T et al. “Annotation of the *M. tuberculosis* hypothetical orfeome: adding functional information to more than half of the uncharacterized proteins”. In: *PLoS ONE* (2012).

[147] Dong, D. et al. “PPE38 modulates the innate immune response and is required for *Mycobacterium marinum* virulence”. In: *Infection and Immunity* (2012).

[148] Dos Vultos, T et al. “DNA repair in *Mycobacterium tuberculosis* revisited”. In: *FEMS Microbiology Reviews* (2009).

[149] Dötsch, A. et al. “The *Pseudomonas aeruginosa* Transcriptional Landscape Is Shaped by Environmental Heterogeneity and Genetic Variation”. In: *mBio* (2015).

[150] Duan, J. et al. “The Complete Genome Sequence of the Plant Growth-Promoting Bacterium Pseudomonas sp. UW4”. In: *PLoS ONE* (2013).

[151] Duque-Ramos, A. et al. “Evaluating the Good Ontology Design Guideline (GoodOD) with the Ontology Quality Requirements and Evaluation Method and Metrics (OQuaRE)”. In: *PLoS ONE* (2014).

[152] Durbin, R. et al. *Biological sequence analysis: probabilistic models of proteins and nucleic acids*. 1998.

[153] Dutilh, B. E. et al. “Explaining microbial phenotypes on a genomic scale: GWAS for microbes”. In: *Briefings in Functional Genomics* (2013).

[154] Eilbeck, K. et al. “The Sequence Ontology: a tool for the unification of genome annotations”. In: *Genome Biology* (2005).

[155] Ekseth, O. K., Kuiper, M., and Mironov, V. “OrthAgogue: an agile tool for the rapid prediction of orthology relations”. In: *Bioinformatics* (2013).

[156] Emmert-Streib, F. “Influence of the experimental design of gene expression studies on the inference of gene regulatory networks: environmental factors”. In: *PeerJ* (2013).

[157] *EmPowerPutida, Exploiting native endowments by re-factoring, re-programming and implementing novel control loops in Pseudomonas putida for bespoke biocatalysis*. 2017. URL: `http://www.empowerputida.eu/`.

[158] ENA. *European Nucleotide Archive Statistics*. 2017. URL: `http://www.ebi.ac.uk/ena/about/statistics/`.

[159] Enright, A. J., Van Dongen, S., and Ouzounis, C. A. “An efficient algorithm for large-scale detection of protein families”. In: *Nucleic Acids Research* (2002).

[160] Fabregat, A. et al. “The Reactome pathway Knowledgebase”. In: *Nucleic Acids Research* (2016).

[161] Faith, J. J. et al. “Large-scale mapping and validation of *Escherichia coli* transcriptional regulation from a compendium of expression profiles”. In: *PLoS Biology* (2007).

[162] Fang, G., Rocha, E., and Danchin, A. “How essential are nonessential genes?” In: *Molecular Biology and Evolution* (2005).

[163] Farhana, A. et al. “Mechanistic insights into a novel exporter-importer system of *Mycobacterium tuberculosis* unravel its role in trafficking of iron”. In: *PLoS ONE* (2008).

[164] Farina, M. et al. “Metals, oxidative stress and neurodegeneration: A focus on iron, manganese and mercury”. In: *Neurochemistry International* (2013).

[165] Federhen, S. et al. “Toward richer metadata for microbial sequences: Replacing strain-level NCBI taxonomy taxids with BioProject, BioSample and Assembly records”. In: *Standards in Genomic Sciences* (2015).

[166] Fernandes, N. D. et al. “A mycobacterial extracytoplasmic sigma factor involved in survival following heat shock and oxidative stress”. In: *Journal of Bacteriology* (1999).

[167] Fernández, J. D. et al. “Binary *RDF* Representation for Publication and Exchange (*HDT*)”. In: *Web Semantics: Science, Services and Agents on the World Wide Web* (2013).

[168] Field, D. et al. “The minimum information about a genome sequences (MIGS) specification”. In: *Nature Biotechnology* (2008).

[169] Finn, R. D. “Pfam: clans, web tools and services”. In: *Nucleic Acids Research* (2006).

[170] Finn, R. D. et al. “InterPro in 2017-beyond protein family and domain annotations”. In: *Nucleic Acids Research* (2017).

[171] Firestine, S. M. et al. “Reactions Catalyzed by 5-Aminoimidazole Ribonucleotide Carboxylases from *Escherichia coli* and *Gallus gallus*: A Case for Divergent Catalytic Mechanisms?” In: *Biochemistry* (1994).

[172] Forbes, J. R. and Gros, P. “Divalent-metal transport by NRAMP proteins at the interface of host-pathogen interactions”. In: *Trends in Microbiology* (2001).

[173] Forbes, J. R. and Gros, P. “Iron, manganese, and cobalt transport by Nramp1 (Slc11a1) and Nramp2 (Slc11a2) expressed at the plasma membrane”. In: *Blood* (2003).

[174] Forrellad, M. A. et al. “Virulence factors of the *Mycobacterium tuberculosis complex*”. In: *Virulence* (2013).

[175] Fortune, S. M. et al. “Mutually dependent secretion of proteins required for mycobacterial virulence”. In: *PNAS* (2005).

[176] Fouts, D. E. et al. “PanOCT: automated clustering of orthologs using conserved gene neighborhood for pan-genomic analysis of bacterial strains and closely related species”. In: *Nucleic Acids Research* (2012).

[177] Francis, R. J., Butler, R. E., and Stewart, G. R. “*Mycobacterium tuberculosis* ESAT-6 is a leukocidin causing Ca2+ influx, necrosis and neutrophil extracellular trap formation”. In: *Cell Death & Disease* (2014).

[178] Friedman, N. et al. “Using Bayesian Networks to Analyze Expression Data”. In: *Journal of Computational Biology* (2004).

[179] Fu, G. et al. “Lsr2 is a nucleoid-associated protein that targets AT-rich sequences and virulence genes in *Mycobacterium tuberculosis*”. In: *PNAS* (2010).

[180] Gabriel, S. E. and Helmann, J. D. “Contributions of Zur-Controlled Ribosomal Proteins to Growth Under Zinc Starvation Conditions”. In: *Journal of Bacteriology* (2009).

[181] Galagan, J. E. et al. “The *Mycobacterium tuberculosis* regulatory network and hypoxia”. In: *Nature* (2013).

[182] Galdzicki, M. et al. “The Synthetic Biology Open Language (SBOL) provides a community standard for communicating designs in synthetic biology”. In: *Nature Biotechnology* (2014).

[183] Gamulin, V, Cetkovic, H, and Ahel, I. “Identification of a promoter motif regulating the major DNA damage response mechanism of *Mycobacterium tuberculosis*”. In: *FEMS Microbiology Letters* (2004).

[184] Garces, A. et al. “EspA Acts as a Critical Mediator of ESX1-Dependent Virulence in *Mycobacterium tuberculosis* by Affecting Bacterial Cell Wall Integrity”. In: *PLoS Pathogens* (2010).

[185] Garlik, S. H., Seaborne, A., and Prud’hommeaux, E. *SPARQL 1.1 Query Language*. 2013. URL: https://www.w3.org/TR/sparql11-query/.

[186] Gatfield, J and Pieters, J. "Essential role for cholesterol in entry of mycobacteria into macrophages". In: *Science* (2000).

[187] Gautam, U. S., Chauhan, S., and Tyagi, J. S. "Determinants outside the DevR C-terminal domain are essential for cooperativity and robust activation of dormancy genes in *Mycobacterium tuberculosis*". In: *PLoS ONE* (2011).

[188] Gazdik, M. A. and Mcdonough, K. A. "Identification of Cyclic AMP-Regulated Genes in *Mycobacterium tuberculosis* Complex Bacteria under Low-Oxygen Conditions". In: *Journal of Bacteriology* (2005).

[189] Gazdik, M. A. et al. "Rv1675c (cmr) regulates intramacrophage and cyclic AMP-induced gene expression in *Mycobacterium tuberculosis*-complex mycobacteria". In: *Molecular Microbiology* (2009).

[190] Generic, T. et al. "The Generic Frame Protocol 2.0 1". In: *Architecture* (1997).

[191] Gengenbacher, M. and Kaufmann, S. H. E. "Mycobacterium tuberculosis: Success through dormancy". In: *FEMS Microbiology Reviews* (2012).

[192] Georgel, P. et al. "A toll-like receptor 2-responsive lipid effector pathway protects mammals against skin infections with gram-positive bacteria". In: *Infection and Immunity* (2005).

[193] Gerasimova, A et al. "Comparative Genomics of the Dormancy Regulons in Mycobacteria". In: *Journal of Bacteriology* (2011).

[194] Giardine, B. et al. "Galaxy: a platform for interactive large-scale genome analysis". In: *Genome Research* (2005).

[195] Giasson, F. and D'Arcus, B. "Bibliographic ontology specification". In: *Biblioteca Nacional Espanola* (2009).

[196] Gill, R, Datta, S, and Datta, S. "A statistical framework for differential network analysis from microarray data". In: *BMC Bioinformatics* (2010).

[197] Glimm, B. et al. "HermiT: An OWL 2 Reasoner". In: *Journal of Automated Reasoning* (2014).

[198] Goecks, J. et al. "Galaxy: a comprehensive approach for supporting accessible, reproducible, and transparent computational research in the life sciences". In: *Genome Biology* (2010).

[199] Goenawan, I. H., Bryan, K., and Lynn, D. J. "DyNet: Visualization and analysis of dynamic molecular interaction networks". In: *Bioinformatics* (2016).

[200] Gomez, J. E. and McKinney, J. D. "*M. tuberculosis* persistence, latency, and drug tolerance". In: *Tuberculosis* (2004).

[201] Gonzalo-Asensio, J. et al. "PhoP: a missing piece in the intricate puzzle of *Mycobacterium tuberculosis* virulence". In: *PLoS ONE* (2008).

[202] Gonzalo-Asensio, J. et al. "The *Mycobacterium tuberculosis* phoPR operon is positively autoregulated in the virulent strain H37Rv". In: *Journal of Bacteriology* (2008).

[203] Gostev, M. et al. "The BioSample Database (BioSD) at the European Bioinformatics Institute". In: *Nucleic Acids Research* (2012).

[204] Gouzy, J, Corpet, F, and Kahn, D. "Whole genome protein domain analysis using a new method for domain clustering". In: *Computers & Chemistry* (1999).

[205] Grant, C. E., Bailey, T. L., and Noble, W. S. "FIMO: scanning for occurrences of a given motif". In: *Bioinformatics* (2011).

[206] Gröschel, M. I. et al. "ESX secretion systems: mycobacterial evolution to counter host immunity". In: *Nature Publishing Group* (2016).

[207] Gross, H. and Loper, J. E. "Genomics of secondary metabolite production by Pseudomonas spp". In: *Natural Product Reports* (2009).

[208] Gross, M. "Antibiotics in crisis". In: *Current Biology* (2013).

[209] Guarino, N., Oberle, D., and Staab, S. “What Is an Ontology?” In: *Handbook on Ontologies*. 2009.

[210] Guirado, E. and Schlesinger, L. S. “Modeling the *Mycobacterium tuberculosis* granuloma – the critical battlefield in host immunity and disease”. In: *Frontiers in Immunology* (2013).

[211] Gupta, A. et al. “*Mycobacterium tuberculosis*: Immune evasion, latency and reactivation”. In: *Immunobiology* (2012).

[212] Gupta, R. K., Srivastava, B. S., and Srivastava, R. “Comparative expression analysis of rpf -like genes of *Mycobacterium tuberculosis H37Rv* under different physiological stress and growth conditions”. In: *Microbiology* (2010).

[213] Gupta, S., Sinha, A., and Sarkar, D. “Transcriptional autoregulation by *Mycobacterium tuberculosis* PhoP involves recognition of novel direct repeat sequences in the regulatory region of the promoter”. In: *FEBS Letters* (2006).

[214] Haak, L. L. et al. “ORCID: a system to uniquely identify researchers”. In: *Learned Publishing* (2012).

[215] Hall, G. et al. “Structure of *Mycobacterium tuberculosis* thioredoxin in complex with quinol inhibitor PMX464”. In: *Protein science* (2011).

[216] Hamilton, J. J. and Reed, J. L. “Software platforms to facilitate reconstructing genome-scale metabolic networks”. In: *Environmental Microbiology* (2014).

[217] Han, X. et al. “Interleukin-10 overexpression in macrophages suppresses atherosclerosis in hyperlipidemic mice”. In: *FASEB Journal* (2010).

[218] Hastie, T et al. *Impute: Imputation for microarray data*. 2010.

[219] Hastings, J. et al. “The ChEBI reference database and ontology for biologically relevant chemistry: enhancements for 2013”. In: *Nucleic Acids Research* (2013).

[220] Hayes, P. J. and Patel-Schneider, P. F. *RDF 1.1 Semantics*. 2014. URL: `https://www.w3.org/TR/rdf11-mt/`.

[221] Heirendt, L. et al. “Creation and analysis of biochemical constraint-based models: the COBRA Toolbox v3.0”. In: *arXiv:1710.04038v2* (2017).

[222] Hendrickx, M. and Leyns, L. “Non-conventional Frizzled ligands and Wnt receptors”. In: *Development Growth & Differentiation* (2008).

[223] Henry, C. S. et al. “iBsu1103: A new genome-scale metabolic model of Bacillus subtilis based on SEED annotations”. In: *Genome Biology* (2009).

[224] Henry, C. S. et al. “High-throughput generation, optimization and analysis of genome-scale metabolic models”. In: *Nature Biotechnology* (2010).

[225] Herrgård, M. J., Fong, S. S., and Palsson, B. “Identification of genome-scale metabolic network models using experimentally measured flux profiles”. In: *PLoS Computational Biology* (2006).

[226] Hoehndorf, R., Schofield, P. N., and Gkoutos, G. V. “The role of ontologies in biological and biomedical research: a functional perspective”. In: *Briefings in Bioinformatics* (2015).

[227] Holmes, B, Willcox, W. R., and Lapage, S. P. “Identification of Enterobacteriaceae by the API 20E system”. In: *Journal of Clinical Pathology* (1978).

[228] Honaker, R. W. et al. “Unique Roles of DosT and DosS in DosR Regulon Induction and *Mycobacterium tuberculosis* Dormancy”. In: *Infection and Immunity* (2009).

[229] Honaker, R. W. et al. “DosS responds to a reduced electron transport system to induce the *Mycobacterium tuberculosis* DosR regulon”. In: *Journal of Bacteriology* (2010).

[230] Horan, K. et al. “Annotating genes of known and unknown function by large-scale coexpression analysis”. In: *Plant Physiology* (2008).

[231] Houben, E. N. G., Korotkov, K. V., and Bitter, W. "Take five - Type VII secretion systems of Mycobacteria". In: *Biochimica et Biophysica Acta* (2013).

[232] Hu, Y. et al. "σE-dependent activation of RbpA controls transcription of the furA-katG operon in response to oxidative stress in mycobacteria". In: *Molecular Microbiology* (2016).

[233] Huttenhower, C et al. "The Sleipnir library for computational functional genomics". In: *Bioinformatics* (2008).

[234] Huynh-Thu, V. A. et al. "Inferring regulatory networks from expression data using tree-based methods". In: *PLoS ONE* (2010).

[235] Hyatt, D. et al. "Prodigal: prokaryotic gene recognition and translation initiation site identification". In: *BMC Bioinformatics* (2010).

[236] Ideker, T. and Krogan, N. J. "Differential network biology". In: *Molecular Systems Biology* (2012).

[237] Ilghari, D. et al. "Solution structure of the *Mycobacterium tuberculosis* EsxG·EsxH complex: functional implications and comparisons with other *M. tuberculosis* Esx family complexes". In: *The Journal of Biological Chemistry* (2011).

[238] Indriate, M and Skaar, E. P. "Nutritional immunity: transition metals at the pathogen-host interface". In: *Nature Reviews Microbiology* (2013).

[239] *INFECT, Systems medicine to understand severe soft tissue infections*. 2017. URL: http://www.fp7infect.eu/.

[240] */inference qualifier*. URL: http://www.insdc.org/.

[241] Ize, B. and Palmer, T. "Mycobacteria's export strategy". In: *Science* (2006).

[242] Jabado, N et al. "Natural resistance to intracellular infections: natural resistance-associated macrophage protein 1 (Nramp1) functions as a pH-dependent manganese transporter at the phagosomal membrane". In: *The Journal of Experimental Medicine* (2000).

[243] Jamwal, S. et al. "Characterizing virulence-specific perturbations in the mitochondrial function of macrophages infected with *Mycobacterium tuberculosis*". In: *Scientific Reports* (2013).

[244] Jamwal, S. V. et al. "Mycobacterial escape from macrophage phagosomes to the cytoplasm represents an alternate adaptation mechanism". In: *Scientific Reports* (2016).

[245] Jang, I. S., Margolin, A, and Califano, A. "hARACNe: Improving the Accuracy of Regulatory Model Reverse Engineering via Higher-order Data Processing Inequality Tests". In: *Interface Focus* (2013).

[246] Jensen, L. J. et al. "STRING 8 - A global view on proteins and their functional interactions in 630 organisms". In: *Nucleic Acids Research* (2009).

[247] Jofré, M. R. et al. "RpoS integrates CRP, Fis, and PhoP signaling pathways to control *Salmonella Typhi* hlyE expression". In: *BMC Microbiology* (2014).

[248] Jones, C. M. et al. "Self-poisoning of *Mycobacterium tuberculosis* by interrupting siderophore recycling". In: *PNAS* (2014).

[249] Jones, P. et al. "InterProScan 5: genome-scale protein function classification". In: *Bioinformatics* (2014).

[250] Jonge, M. I. de et al. "ESAT-6 from *Mycobacterium tuberculosis* dissociates from its putative chaperone CFP-10 under acidic conditions and exhibits membrane-lysing activity". In: *Journal of Bacteriology* (2007).

[251] Joseph, S. et al. "*Mycobacterium tuberculosis* Cpn60.2 (GroEL2) blocks macrophage apoptosis via interaction with mitochondrial mortalin". In: *Biology Open* (2017).

[252] Joseph, S. V. et al. "Comparative analysis of mycobacterial truncated hemoglobin promoters and the groEL2 promoter in free-living and intracellular mycobacteria". In: *Applied and Environmental Microbiology* (2012).

[253] Jung, J.-Y. et al. "The intracellular environment of human macrophages that produce nitric oxide promotes growth of mycobacteria". In: *Infection and Immunity* (2013).

[254] Jupp, S. et al. "The EBI RDF platform: linked open data for the life sciences". In: *Bioinformatics* (2014).

[255] Juttukonda, L. J. and Skaar, E. P. "Manganese homeostasis and utilization in pathogenic bacteria". In: *Molecular Microbiology* (2015).

[256] Kahramanoglou, C. et al. "Genomic mapping of cAMP receptor protein (CRPMt) in *Mycobacterium tuberculosis*: Relation to transcriptional start sites and the role of CRPMt as a transcription factor". In: *Nucleic Acids Research* (2014).

[257] Kalamidas, S. a. et al. "cAMP synthesis and degradation by phagosomes regulate actin assembly and fusion events: consequences for mycobacteria". In: *Journal of Cell Science* (2006).

[258] Kamminga, T. et al. "Persistence of Functional Protein Domains in Mycoplasma Species and their Role in Host Specificity and Synthetic Minimal Life". In: *Frontiers in Cellular and Infection Microbiology* (2017).

[259] Kanehisa, M et al. "KEGG for integration and interpretation of large-scale molecular data sets". In: *Nucleic Acids Research* (2012).

[260] Kanehisa, M. et al. "KEGG as a reference resource for gene and protein annotation". In: *Nucleic Acids Research* (2016).

[261] Kang, S.-M. et al. "Functional details of the *Mycobacterium tuberculosis* VapBC26 toxin-antitoxin system based on a structural study: insights into unique binding and antibiotic peptides". In: *Nucleic Acids Research* (2017).

[262] Kapoor, N. et al. "Human granuloma in vitro model, for TB dormancy and resuscitation". In: *PLoS ONE* (2013).

[263] Karakousis, P. C., Williams, E. P., and Bishai, W. R. "Altered expression of isoniazid-regulated genes in drug-treated dormant *Mycobacterium tuberculosis*". In: *Journal of Antimicrobial Chemotherapy* (2008).

[264] Karp, P. D. et al. "Pathway tools version 13.0: integrated software for pathway/genome informatics and systems biology". In: *Briefings in Bioinformatics* (2009).

[265] Kaur, K. et al. "DevS/DosS sensor is bifunctional and its phosphatase activity precludes aerobic DevR/DosR regulon expression in *Mycobacterium tuberculosis*". In: *FEBS Journal* (2016).

[266] Kelder, T. et al. "WikiPathways: Building research communities on biological pathways". In: *Nucleic Acids Research* (2012).

[267] Khan, A. and Sarkar, D. "Nitrate reduction pathways in mycobacteria and their implications during latency". In: *Microbiology* (2012).

[268] Khare, G., Nangpal, P., and Tyagi, A. K. "Differential roles of iron storage proteins in maintaining the iron homeostasis in *Mycobacterium tuberculosis*". In: *PLoS ONE* (2017).

[269] King, Z. A. et al. "BiGG Models: A platform for integrating, standardizing and sharing genome-scale models". In: *Nucleic Acids Research* (2016).

[270] Kitano, H. "Systems biology: A brief overview". In: *Science* (2002).

[271] Koehorst, J. J. et al. "Protein domain architectures provide a fast, efficient and scalable alternative to sequence-based methods for comparative functional genomics [version 1; referees: 1 approved, 2 approved with reservations]". In: *F1000Research* (2016).

[272] Koehorst, J. J. et al. "Comparison of 432 Pseudomonas strains through integration of genomic, functional, metabolic and expression data". In: *Scientific Reports* (2016).

[273] Koehorst, J. J. et al. "SAPP: functional genome annotation and analysis through a semantic framework using FAIR principles". In: *Bioinformatics* (2018).

[274] Korch, S. B., Contreras, H., and Clark-curtiss, J. E. "Three *Mycobacterium tuberculosis* Rel Toxin-Antitoxin Modules Inhibit Mycobacterial Growth and Are Expressed in Infected Human Macrophages". In: *Journal of Bacteriology* (2009).

[275] Köser, C. U., Ellington, M. J., and Peacock, S. J. "Whole-genome sequencing to control antimicrobial resistance". In: *Trends in Genetics* (2014).

[276] Kreuzer, K. N. "DNA damage responses in prokaryotes: Regulating gene expression, modulating growth patterns, and manipulating replication forks". In: *Cold Spring Harbor Perspectives in Biology* (2013).

[277] Kumar, A. et al. "*Mycobacterium tuberculosis* DosS is a redox sensor and DosT is a hypoxia sensor". In: *PNAS* (2007).

[278] Kumar, A. et al. "Heme oxygenase-1-derived carbon monoxide induces the *Mycobacterium tuberculosis* dormancy regulon". In: *The Journal of Biological Chemistry* (2008).

[279] Kumar, A. et al. "Redox homeostasis in mycobacteria: the key to tuberculosis control?" In: *Expert Reviews in Molecular Medicine* (2011).

[280] Kumar, V. A. et al. "EspR-dependent ESAT-6 Protein Secretion of *Mycobacterium tuberculosis* Requires the Presence of Virulence Regulator PhoP". In: *Journal of Biological Chemistry* (2016).

[281] Kumar, V. S. and Maranas, C. D. "GrowMatch: An automated method for reconciling in silico/in vivo growth predictions". In: *PLoS Computational Biology* (2009).

[282] Kurthkoti, K. et al. "The Capacity of *Mycobacterium tuberculosis* To Survive Iron Starvation Might Enable It To Persist in Iron-Deprived Microenvironments of Human Granulomas". In: *mBio* (2017).

[283] Kutmon, M. et al. "WikiPathways: Capturing the full diversity of pathway knowledge". In: *Nucleic Acids Research* (2016).

[284] Kwong, J. C. et al. "Whole genome sequencing in clinical and public health microbiology". In: *Pathology* (2015).

[285] Lam, S. D. et al. "Gene3D: Expanding the utility of domain assignments". In: *Nucleic Acids Research* (2016).

[286] Langfelder, P and Horvath, S. "WGCNA: an R package for weighted correlation network analysis". In: *BMC Bioinformatics* (2008).

[287] Larsen, S. J. and Baumbach, J. "CytoMCS: A Multiple Maximum Common Subgraph Detection Tool for Cytoscape". In: *Journal of Integrative Bioinformatics* (2017).

[288] Larsson, C. et al. "Gene expression of *Mycobacterium tuberculosis* putative transcription factors whiB1-7 in redox environments". In: *PLoS ONE* (2012).

[289] Latendresse, M. "Efficiently gap-filling reaction networks". In: *BMC Bioinformatics* (2014).

[290] Law, V. et al. "DrugBank 4.0: shedding new light on drug metabolism". In: *Nucleic Acids Research* (2014).

[291] Lebo, T., Sahoo, S., and McGuinness, D. "PROV-O: The PROV Ontology". In: *W3C Recommendation* (2013).

[292] Lee, H.-N. et al. "Protein-protein interactions between histidine kinases and response regulators of *Mycobacterium tuberculosis H37Rv*". In: *Journal of Microbiology* (2012).

[293] Lee, I.-g. et al. “Structural and functional studies of the *Mycobacterium tuberculosis* VapBC30 toxin-antitoxin system: implications for the design of novel antimicrobial peptides”. In: *Nucleic Acids Research* (2015).

[294] Lee, S. A. et al. “General and condition-specific essential functions of Pseudomonas aeruginosa”. In: *Proceedings of the National Academy of Sciences of the United States of America of the United States of America* (2015).

[295] Leistikow, R. L. et al. “The *Mycobacterium tuberculosis* DosR regulon assists in metabolic homeostasis and enables rapid recovery from nonrespiring dormancy”. In: *Journal of Bacteriology* (2010).

[296] Leopold, S. R. et al. “Bacterial whole-genome sequencing revisited: Portable, scalable, and standardized analysis for typing and detection of virulence and antibiotic resistance genes”. In: *Journal of Clinical Microbiology* (2014).

[297] Lew, J. M. et al. “TubercuList–10 years after”. In: *Tuberculosis* (2011).

[298] Li, L., Stoeckert, C. J., and Roos, D. S. “OrthoMCL: identification of ortholog groups for eukaryotic genomes”. In: *Genome Research* (2003).

[299] Li, W. et al. “*Mycobacterium tuberculosis* Rv3402c Enhances Mycobacterial Survival within Macrophages and Modulates the Host Pro-Inflammatory Cytokines Production via NF-Kappa B/ERK/p38 Signaling”. In: *PLoS ONE* (2014).

[300] Liberati, N. T. et al. “An ordered, nonredundant library of *Pseudomonas aeruginosa* strain PA14 transposon insertion mutants”. In: *Proceedings of the National Academy of Sciences of the United States of America of the United States of America* (2006).

[301] Lim, Y.-J. et al. “*Mycobacterium kansasii*-induced death of murine macrophages involves endoplasmic reticulum stress responses mediated by reactive oxygen species generation or calpain activation”. In: *Apoptosis* (2013).

[302] Lin, H., Andersen, G. R., and Yatime, L. “Crystal structure of human S100A8 in complex with zinc and calcium”. In: *BMC Structural Biology* (2016).

[303] Litwin, C. M. and Calderwood, S. B. “Role of Iron in Regulation of Virulence Genes”. In: *Clinical Microbiology Reviews* (1993).

[304] Loper, J. E. et al. “Comparative genomics of plant-associated Pseudomonas spp.: insights into diversity and inheritance of traits involved in multitrophic interactions”. In: *PLoS Genetics* (2012).

[305] Los, F. C. O. et al. “Role of pore-forming toxins in bacterial infectious diseases”. In: *Microbiology and Molecular Biology Reviews* (2013).

[306] Lou, Y. et al. “EspC forms a filamentous structure in the cell envelope of *Mycobacterium tuberculosis* and impacts ESX-1 secretion”. In: *Molecular Microbiology* (2017).

[307] Lucarelli, D. et al. “The Metal-Dependent Regulators FurA and FurB from *Mycobacterium Tuberculosis*”. In: *International Journal of Molecular Sciences* (2008).

[308] Luo, M., Fadeev, E. A., and Groves, J. T. “Mycobactin-mediated iron acquisition within macrophages”. In: *Nature Chemical Biology* (2005).

[309] M. Sritharan. “Iron Homeostasis in *Mycobacterium tuberculosis*: Mechanistic Insights into Siderophore-Mediated Iron Uptake”. In: *Bacteriology* (2016).

[310] Ma, J. et al. “MRFalign: Protein Homology Detection through Alignment of Markov Random Fields”. In: *PLoS Computational Biology* (2014).

[311] Ma, W and Wong, W. H. “Chapter Three - The Analysis of ChIP-Seq Data”. In: *Methods in Enzymology* (2011).

[312] Maciag, A. et al. “Global analysis of the *Mycobacterium tuberculosis* Zur (FurB) regulon”. In: *Journal of Bacteriology* (2007).

[313] Madeira, S. C. and Oliveira, A. L. "Biclustering Algorithms for Biological Data Analysis: A Survey". In: *IEEE/ACM Transactions on Computational Biology and Bioinformatics* (2004).

[314] Madigan, C. A. et al. "Lipidomic discovery of deoxysiderophores reveals a revised mycobactin biosynthesis pathway in *Mycobacterium tuberculosis*". In: *PNAS* (2012).

[315] Maere, S., Heymans, K., and Kuiper, M. "BiNGO: A Cytoscape plugin to assess overrepresentation of Gene Ontology categories in Biological Networks". In: *Bioinformatics* (2005).

[316] Mahajan, S. et al. "*Mycobacterium tuberculosis* modulates macrophage lipid-sensing nuclear receptors PPARγ and TR4 for survival". In: *Journal of Immunology* (2012).

[317] Manabe, Y. C. et al. "Attenuation of virulence in *Mycobacterium tuberculosis* expressing a constitutively active iron repressor". In: *Proceedings of the National Academy of Sciences of the United States of America of the United States of America* (1999).

[318] Manganelli, R. and Provvedi, R. "An integrated regulatory network including two positive feedback loops to modulate the activity of SigE in mycobacteria". In: *Molecular Microbiology* (2010).

[319] Marbach, D. et al. "Wisdom of crowds for robust gene network inference". In: *Nature Methods* (2012).

[320] Margolin, A. A. et al. "ARACNE: An algorithm for the reconstruction of gene regulatory networks in a mammalian cellular context". In: *BMC Bioinformatics* (2006).

[321] Marino, S., El-Kebir, M., and Kirschner, D. "A hybrid multi-compartment model of granuloma formation and T cell priming in tuberculosis". In: *Journal of Theoretical Biology* (2011).

[322] Maris, A. E. et al. "Dimerization allows DNA target site recognition by the NarL response regulator". In: *Nature Structural Biology* (2002).

[323] Marquart, H. V. et al. "C1q deficiency in an Inuit family: Identification of a new class of C1q disease-causing mutations". In: *Clinical Immunology* (2007).

[324] Marti, T. M., Kunz, C, and Fleck, O. "DNA mismatch repair and mutation avoidance pathways". In: *Journal of Cellular Physiology* (2002).

[325] Matange, N. "Revisiting bacterial cyclic nucleotide phosphodiesterases: cyclic AMP hydrolysis and beyond". In: *FEMS Microbiology Letters* (2015).

[326] Matthews, B. W. "Comparison of the predicted and observed secondary structure of T4 phage lysozyme". In: *Biochimica et Biophysica Acta - Protein Structure* (1975).

[327] McCarthy, L., Vandervalk, B., and Wilkinson, M. "SPARQL assist language-neutral query composer". In: *BMC Bioinformatics* (2012).

[328] McMahon, M. D., Rush, J. S., and Thomas, M. G. "Analyses of MbtB, MbtE, and MbtF suggest revisions to the mycobactin biosynthesis pathway in Mycobacterium tuberculosis". In: *Journal of Bacteriology* (2012).

[329] Medini, D. et al. "The microbial pan-genome". In: *Current Opinion in Genetics & Development* (2005).

[330] Meena, L. S. and Rajni. "Survival mechanisms of pathogenic *Mycobacterium tuberculosis H37Rv*". In: *FEBS Journal* (2010).

[331] Mehra, A. et al. "Mycobacterium tuberculosis Type VII Secreted Effector EsxH Targets Host ESCRT to Impair Trafficking". In: *PLoS Pathogens* (2013).

[332] Mehra, S. et al. "The DosR Regulon Modulates Adaptive Immunity and Is Essential for *Mycobacterium tuberculosis* Persistence". In: *American Journal of Respiratory and Critical Care Medicine* (2015).

[333] Mele, T. and Madrenas, J. "TLR2 signalling: At the crossroads of commensalism, invasive infections and toxic shock syndrome by *Staphylococcus aureus*". In: *International Journal of Biochemistry and Cell Biology* (2010).

[334] Meng, L. et al. "PPE38 Protein of *Mycobacterium tuberculosis* Inhibits Macrophage MHC Class I Expression and Dampens CD8+ T Cell Responses". In: *Frontiers in Cellular and Infection Microbiology* (2017).

[335] Meng, Q. et al. "Systems Biology Approaches and Applications in Obesity, Diabetes, and Cardiovascular Diseases". In: *Current Cardiovascular Risk Reports* (2013).

[336] Menge, B. A. et al. "Selective amino acid deficiency in patients with impaired glucose tolerance and type 2 diabetes". In: *Regulatory Peptides* (2010).

[337] Meyer, P. E., Lafitte, F, and Bontempi, G. "minet: A R/Bioconductor Package for Inferring Large Transcriptional Networks Using Mutual Information". In: *BMC Bioinformatics* (2008).

[338] Miles, A. et al. "SKOS Core: Simple knowledge organisation for the Web". In: *International Conference on Dublin Core and Metadata Applications* (2005).

[339] Mitchell, A. et al. "The InterPro protein families database: The classification resource after 15 years". In: *Nucleic Acids Research* (2015).

[340] Mitraka, E. et al. "Wikidata: A platform for data integration and dissemination for the life sciences and beyond". In: *bioRxiv* (2015).

[341] Mons, B. et al. "The value of data". In: *Nature Genetics* (2011).

[342] Montecchi-Palazzi, L. et al. "The PSI-MOD community standard for representation of protein modification data". In: *Nature Biotechnology* (2008).

[343] Moretti, S. et al. "MetaNetX/MNXref - Reconciliation of metabolites and biochemical reactions to bring together genome-scale metabolic networks". In: *Nucleic Acids Research* (2016).

[344] Morgulis, A et al. "Database indexing for production MegaBLAST searches". In: *Bioinformatics* (2008).

[345] Mosquera-Rendón, J. et al. "Pangenome-wide and molecular evolution analyses of the Pseudomonas aeruginosa species". In: *BMC Genomics* (2016).

[346] Mowa, M. B. et al. "Function and Regulation of Class I Ribonucleotide Reductase-Encoding Genes in Mycobacteria". In: *Journal of Bacteriology* (2009).

[347] Musen, M. A. "The protégé project: a look back and a look forward". In: *AI Matters* (2015).

[348] *MycoSynVac, Engineering Mycoplasma pneumoniae as a broad-spectrum animal vaccine*. 2017. URL: `http://www.mycosynvac.eu/`.

[349] Naffin-Olivos, J. L. et al. "*Mycobacterium tuberculosis* Hip1 Modulates Macrophage Responses through Proteolysis of GroEL2". In: *PLoS Pathogens* (2014).

[350] Nambu, S. et al. "A new way to degrade heme: the *Mycobacterium tuberculosis* enzyme MhuD catalyzes heme degradation without generating CO". In: *The Journal of Biological Chemistry* (2013).

[351] Nawrocki, J. and Wojciechowski, A. "Experimental evaluation of pair programming". In: *European Software Control and Metrics* (2001).

[352] Nelson, K. E. et al. "Complete genome sequence and comparative analysis of the metabolically versatile Pseudomonas putida KT2440". In: *Environmental Microbiology* (2002).

[353] Nguyen, L. and Pieters, J. “Mycobacterial subversion of chemotherapeutic reagents and host defense tactics: challenges in tuberculosis drug development”. In: *Annual Review of Pharmacology and Toxicology* (2009).

[354] Nguyen, N.-N. et al. “EnzDP: Improved enzyme annotation for metabolic network reconstruction based on domain composition profiles”. In: *Journal of Bioinformatics and Computational Biology* (2015).

[355] Nguyen, V., Bodenreider, O., and Sheth, A. “Don’t Like RDF Reification? Making Statements about Statements Using Singleton Property”. In: *Proceedings of the International World-Wide Web Conference* (2014).

[356] Nogales, J., Palsson, B., and Thiele, I. “A genome-scale metabolic reconstruction of Pseudomonas putida KT2440: iJN746 as a cell factory”. In: *BMC Systems Biology* (2008).

[357] Notebaart, R. A. et al. “Correlation between sequence conservation and the genomic context after gene duplication”. In: *Nucleic Acids Research* (2005).

[358] Oberhardt, M. A., Palsson, B., and Papin, J. A. “Applications of genome-scale metabolic reconstructions”. In: *Molecular Systems Biology* (2009).

[359] Oberhardt, M. a. et al. “Genome-Scale Metabolic Network Analysis of the Opportunistic Pathogen Pseudomonas aeruginosa PAO1”. In: *Journal of Bacteriology* (2008).

[360] Oberhardt, M. a. et al. “Reconciliation of genome-scale metabolic reconstructions for comparative systems analysis”. In: *PLoS Computational Biology* (2011).

[361] O’Brien, E. J., Monk, J. M., and Palsson, B. O. “Using genome-scale models to predict biological capabilities”. In: *Cell* (2015).

[362] Oh, Y. K. et al. “Genome-scale reconstruction of metabolic network in Bacillus subtilis based on high-throughput phenotyping and gene essentiality data”. In: *Journal of Biological Chemistry* (2007).

[363] Olakanmi, O. et al. “Intraphagosomal *Mycobacterium tuberculosis* acquires iron from both extracellular transferrin and intracellular iron pools. Impact of interferon-gamma and hemochromatosis”. In: *The Journal of Biological Chemistry* (2002).

[364] Olakanmi, O. et al. “The Nature of Extracellular Iron Influences Iron Acquisition by *Mycobacterium tuberculosis* Residing within Human Macrophages”. In: *Infection and Immunity* (2004).

[365] Oldridge, D. A. et al. “The mycobacterial iron dependent regulator IdeR induces ferritin (bfrB) by alleviating Lsr2 repression”. In: *Molecular Microbiology* (2016).

[366] OpenLink Software Inc. *iSPARQL*. 2007. URL: `https://github.com/openlink/iSPARQL`.

[367] Orth, J. D., Thiele, I., and Palsson, B. O. “What is flux balance analysis?” In: *Nature Biotechnology* (2010).

[368] Orth, J. D. et al. “A comprehensive genome-scale reconstruction of *Escherichia coli* metabolism”. In: *Molecular Systems Biology* (2011).

[369] Otero, J. M. and Nielsen, J. “Industrial systems biology”. In: *Biotechnology and Bioengineering* (2010).

[370] Outten, F. W. and Theil, E. C. “Iron-based redox switches in biology”. In: *Antioxidants & Redox Signaling* (2009).

[371] Overbeek, R. et al. “The SEED and the Rapid Annotation of microbial genomes using Subsystems Technology (RAST)”. In: *Nucleic Acids Research* (2014).

[372] Page, A. J. et al. “Roary: rapid large-scale prokaryote pan genome analysis”. In: *Bioinformatics* (2015).

[373] Paige, C. and Bishai, W. R. "Penitentiary or penthouse condo: the tuberculous granuloma from the microbe's point of view". In: *Cellular Microbiology* (2010).

[374] Pandey, R. and Rodriguez, G. M. "IdeR is required for iron homeostasis and virulence in *Mycobacterium tuberculosis*". In: *Molecular Microbiology* (2014).

[375] Pandey, R. et al. "MntR(Rv2788): a transcriptional regulator that controls manganese homeostasis in M ycobacterium tuberculosis". In: *Molecular Microbiology* (2015).

[376] Pandey, S. D. et al. "Iron-regulated protein HupB of *Mycobacterium tuberculosis* positively regulates siderophore biosynthesis and is essential for growth in macrophages". In: *Journal of Bacteriology* (2014).

[377] Pang, X. et al. "Evidence for Complex Interactions of Stress-Associated Regulons in an mprAB Deletion Mutant of *Mycobacterium Tuberculosis*". In: *Microbiology* (2007).

[378] Pang, X. et al. "The β-propeller gene Rv1057 of *Mycobacterium tuberculosis* has a complex promoter directly regulated by both the MprAB and TrcRS two-component systems". In: *Tuberculosis* (2011).

[379] Pang, X. et al. "MprAB regulates the espA operon in *Mycobacterium tuberculosis* and modulates ESX-1 function and host cytokine response". In: *Journal of Bacteriology* (2013).

[380] Papp-Wallace, K. M. and Maguire, M. E. "Manganese transport and the role of manganese in virulence". In: *Annual Review of Microbiology* (2006).

[381] Park, H.-D. et al. "Rv3133c/dosR is a transcription factor that mediates the hypoxic response of *Mycobacterium tuberculosis*". In: *Molecular Microbiology* (2003).

[382] Pasek, S., Risler, J.-L., and Brézellec, P. "Gene fusion/fission is a major contributor to evolution of multi-domain bacterial proteins". In: *Bioinformatics* (2006).

[383] Pavlopoulos, G. A. et al. "Using graph theory to analyze biological networks". In: *BioData Mining* (2011).

[384] Pechter, K. B. et al. "Two roles for aconitase in the regulation of tricarboxylic acid branch gene expression in Bacillus subtilis". In: *Journal of Bacteriology* (2013).

[385] Peracino, B., Buracco, S., and Bozzaro, S. "The Nramp (Slc11) proteins regulate development, resistance to pathogenic bacteria and iron homeostasis in Dictyostelium discoideum". In: *Journal of Cell Science* (2013).

[386] Pereira, C. P., Bachli, E. B., and Schoedon, G. "The Wnt pathway: A macrophage effector molecule that triggers inflammation". In: *Current Atherosclerosis Reports* (2009).

[387] Pérez, E et al. "An essential role for phoP in *Mycobacterium tuberculosis* virulence". In: *Molecular Microbiology* (2001).

[388] Perez-Urbina, H., Siren, E., and Clark, K. *Validating Semantic Web Data with OWL Integrity Constraints*. 2010. URL: `https://www.stardog.com/docs/4.1.3/icv/icv-specification`.

[389] Petersen, T. N. et al. "SignalP 4.0: discriminating signal peptides from transmembrane regions". In: *Nature Methods* (2011).

[390] Phetsuksiri, B. et al. "Antimycobacterial Activities of Isoxyl and New Derivatives through the Inhibition of Mycolic Acid Synthesis". In: *Antimicrobial Agents and Chemotherapy* (1999).

[391] Pohl, E, Holmes, R. K., and Hol, W. G. "Crystal structure of the iron-dependent regulator (IdeR) from *Mycobacterium tuberculosis* shows both metal binding sites fully occupied". In: *Journal of Molecular Biology* (1999).

[392] Popov, I. O. et al. "Connecting the Dots: A Multi-pivot Approach to Data Exploration". In: *The Semantic Web – International Semantic Web Conference* (2011).

[393] Poveda-Villalón, M., Gómez-Pérez, A., and Suárez-Figueroa, M. C. "OOPS! (OntOlogy Pitfall Scanner!): An On-line Tool for Ontology Evaluation". In: *International Journal on Semantic Web and Information Systems* (2014).

[394] Prados-Rosales, R. et al. "Role for *mycobacterium tuberculosis* membrane vesicles in iron acquisition". In: *Journal of Bacteriology* (2014).

[395] Prud'hommeaux, E., Labra Gayo, J. E., and Solbrig, H. "Shape expressions: an RDF validation and transformation language". In: *Proceedings of the 10th International Conference on Semantic Systems* (2014).

[396] Prud'hommeaux, E. and Seaborne, A. *SPARQL Query Language for RDF*. 2008. URL: `http://www.w3.org/TR/rdf-sparql-query/`.

[397] Puchalka, J et al. "Genome-Scale Reconstruction and Analysis of the Pseudomonas putida KT2440 Metabolic Network Facilitates Applications in Biotechnology". In: *PLoS Computational Biology* (2008).

[398] Qu, K. et al. "Integrative genomic analysis by interoperation of bioinformatics tools in GenomeSpace". In: *Nature Methods* (2016).

[399] R Core Team. *R: A language and environment for statistical computing*. 2017.

[400] Rabiner, L. and Juang, B. "An introduction to hidden Markov models". In: *IEEE ASSP Magazine* (1986).

[401] Raghunand, T. R. and Bishai, W. R. "*Mycobacterium smegmatis* whmD and its homologue *Mycobacterium tuberculosis* whiB2 are functionally equivalent". In: *Microbiology* (2006).

[402] Raman, K., Bhat, A. G., and Chandra, N. "A systems perspective of host-pathogen interactions: predicting disease outcome in tuberculosis". In: *Molecular Biosystems* (2010).

[403] Ranganathan, S. et al. "Characterization of a cAMP responsive transcription factor, Cmr (Rv1675c), in TB complex mycobacteria reveals overlap with the DosR (DevR) dormancy regulon". In: *Nucleic Acids Research* (2002).

[404] Ravasz, E et al. "Hierarchical organization of modularity in metabolic networks". In: *Science* (2002).

[405] Reddy, S. K. et al. "Eukaryotic-like adenylyl cyclases in *Mycobacterium tuberculosis H37Rv*: cloning and characterization". In: *The Journal of Biological Chemistry* (2001).

[406] Reddy, T. B. K. et al. "TB database: an integrated platform for tuberculosis research". In: *Nucleic Acids Research* (2009).

[407] Reed, J. L. et al. "Systems approach to refining genome annotation". In: *Proceedings of the National Academy of Sciences of the United States of America of the United States of America* (2006).

[408] Reeves, A. Z. et al. "Aminoglycoside cross-resistance in *Mycobacterium tuberculosis* due to mutations in the 5′ untranslated region of whiB7". In: *Antimicrobial Agents and Chemotherapy* (2013).

[409] Reiss, D. J., Baliga, N. S., and Bonneau, R. "Integrated biclustering of heterogeneous genome-wide datasets for the inference of global regulatory networks". In: *BMC Bioinformatics* (2006).

[410] Reshef, D. N. et al. "Detecting Novel Associations in Large Data Sets". In: *Science* (2011).

[411] Rickman, L. et al. "A member of the cAMP receptor protein family of transcription regulators in Mycobacterium tuberculosis is required for virulence in mice and controls transcription of the rpfA gene coding for a resuscitation promoting factor". In: *Molecular Microbiology* (2005).

[412] Rienksma, R. A. et al. "Comprehensive insights into transcriptional adaptation of intracellular mycobacteria by microbe-enriched dual RNA sequencing". In: *BMC Genomics* (2015).

[413] Rishi, P. et al. "Salmonella-macrophage interactions upon manganese supplementation". In: *Biological Trace Element Research* (2010).

[414] Roback, P et al. "A predicted operon map for *Mycobacterium tuberculosis*". In: *Nucleic Acids Research* (2007).

[415] Rocca-Serra, P. et al. "ISA software suite: Supporting standards-compliant experimental annotation and enabling curation at the community level". In: *Bioinformatics* (2010).

[416] Rodriguez, G. M. and Smith, I. "Mechanisms of iron regulation in mycobacteria: role in physiology and virulence". In: *Molecular Microbiology* (2003).

[417] Rodriguez, G. M. et al. "ideR, an Essential Gene in *Mycobacterium tuberculosis*: Role of IdeR in Iron-Dependent Gene Expression, Iron Metabolism, and Oxidative Stress Response". In: *Infection and Immunity* (2002).

[418] Rohde, K. H., Abramovitch, R. B., and Russell, D. G. "*Mycobacterium tuberculosis* invasion of macrophages: linking bacterial gene expression to environmental cues". In: *Cell Host & Microbe* (2007).

[419] Rohde, K. H. et al. "Linking the transcriptional profiles and the physiological states of *Mycobacterium tuberculosis* during an extended intracellular infection". In: *PLoS Pathogens* (2012).

[420] Rosselló-Móra, R. and Amann, R. "Past and future species definitions for Bacteria and Archaea". In: *Systematic and Applied Microbiology* (2015).

[421] Rowley, J. "The wisdom hierarchy: Representations of the DIKW hierarchy". In: *Journal of Information Science* (2007).

[422] Russell, D. G. et al. "Foamy macrophages and the progression of the human TB granuloma". In: *Nature Immunology* (2010).

[423] Russell, S & Norvig, P. *Artificial intelligence: a modern approach*. 2009.

[424] Rustad, T. R. et al. "The enduring hypoxic response of *Mycobacterium tuberculosis*". In: *PLoS ONE* (2008).

[425] Saccenti, E. et al. "Probabilistic networks of blood metabolites in healthy subjects as indicators of latent cardiovascular risk". In: *Journal of Proteome Research* (2015).

[426] Sachdeva, P et al. "The sigma factors of *Mycobacterium tuberculosis*:regulation of the regulators". In: *FEBS Journal* (2010).

[427] Saini, V., Farhana, A., and Steyn, A. J. C. "*Mycobacterium tuberculosis* WhiB3: a novel iron-sulfur cluster protein that regulates redox homeostasis and virulence". In: *Antioxidants & Redox Signaling* (2012).

[428] Samuel, L. P. et al. "Expression, production and release of the Eis protein by *Mycobacterium tuberculosis* during infection of macrophages and its effect on cytokine secretion". In: *Microbiology* (2007).

[429] Sani, M. et al. "Direct Visualization by Cryo-EM of the Mycobacterial Capsular Layer: A Labile Structure Containing ESX-1-Secreted Proteins". In: *PLoS Pathogens* (2010).

[430] Sanyal, S. et al. "Polyphosphate kinase 1, a central node in the stress response network of *Mycobacterium tuberculosis*, connects the two-component systems MprAB and SenX3-RegX3 and the extracytoplasmic function sigma factor, sigma E". In: *Microbiology* (2013).

[431] Sanz, J et al. "The Transcriptional Regulatory Network of *Mycobacterium tuberculosis*". In: *PLoS ONE* (2011).

[432] Satish Kumar, V., Dasika, M. S., and Maranas, C. D. “Optimization based automated curation of metabolic reconstructions”. In: *BMC Bioinformatics* (2007).

[433] Schaible, U. E. and Kaufmann, S. H. E. “Iron and microbial infection”. In: *Nature Reviews Microbiology* (2004).

[434] Schellenberger, J. et al. “Quantitative prediction of cellular metabolism with constraint-based models: the COBRA Toolbox v2. 0”. In: *Nature Protocols* (2011).

[435] Schomburg, I. et al. “BRENDA, the enzyme database: updates and major new developments”. In: *Nucleic Acids Research* (2004).

[436] Schweiger, D., Trajanoski, Z., and Pabinger, S. “SPARQLGraph: a web-based platform for graphically querying biological Semantic Web databases”. In: *BMC Bioinformatics* (2014).

[437] Seemann, T. “Prokka: rapid prokaryotic genome annotation”. In: *Bioinformatics* (2014).

[438] Serafini, A. et al. “Characterization of a *Mycobacterium tuberculosis* ESX-3 Conditional Mutant: Essentiality and Rescue by Iron and Zinc”. In: *Journal of Bacteriology* (2009).

[439] Serafini, A. et al. “The ESX-3 Secretion System Is Necessary for Iron and Zinc Homeostasis in *Mycobacterium tuberculosis*”. In: *PLoS ONE* (2013).

[440] Sethi, D. et al. “Lipoprotein LprI of *Mycobacterium tuberculosis* Acts as a Lysozyme Inhibitor *”. In: *The Journal of Biological Chemistry* (2016).

[441] Seto, S., Tsujimura, K., and Koide, Y. “Rab GTPases regulating phagosome maturation are differentially recruited to mycobacterial phagosomes”. In: *Traffic* (2011).

[442] Shaler, C. R. et al. “Within the Enemy’s Camp: contribution of the granuloma to the dissemination, persistence and transmission of Mycobacterium tuberculosis”. In: *Frontiers in Immunology* (2013).

[443] Shannon, P. et al. “Cytoscape: A software Environment for integrated models of biomolecular interaction networks”. In: *Genome Research* (2003).

[444] Shannon, P. et al. “Cytoscape: A software Environment for integrated models of biomolecular interaction networks”. In: *Genome Research* (2003).

[445] Sharma, S. and Tyagi, J. S. “*Mycobacterium tuberculosis* DevR/DosR dormancy regulator activation mechanism: Dispensability of phosphorylation, cooperativity and essentiality of α10 Helix”. In: *PLoS ONE* (2016).

[446] Shea, A. et al. “Biolog phenotype microarrays”. In: *Methods in Molecular Biology* (2012).

[447] Siegrist, M. S. et al. “Mycobacterial Esx-3 is required for mycobactin-mediated iron acquisition”. In: *PNAS* (2009).

[448] Silva-Gomes, S. et al. “Heme catabolism by heme oxygenase-1 confers host resistance to Mycobacterium infection”. In: *Infection and Immunity* (2013).

[449] Silva Miranda, M. et al. “The tuberculous granuloma: an unsuccessful host defence mechanism providing a safety shelter for the bacteria?” In: *Clinical & Developmental Immunology* (2012).

[450] Silvester, N. et al. “The European Nucleotide Archive in 2017”. In: *Nucleic Acids Research* (2018).

[451] Simeone, R., Bottai, D., and Brosch, R. “ESX/type VII secretion systems and their role in host-pathogen interaction”. In: *Current Opinion in Microbiology* (2009).

[452] Simeone, R. et al. “Phagosomal rupture by *Mycobacterium tuberculosis* results in toxicity and host cell death”. In: *PLoS Pathogens* (2012).

[453] Singh, A. et al. "*Mycobacterium tuberculosis* WhiB3 responds to O2 and nitric oxide via its [4Fe-4S] cluster and is essential for nutrient starvation survival". In: *PNAS* (2007).

[454] Singh, A. et al. "*Mycobacterium tuberculosis* WhiB3 maintains redox homeostasis by regulating virulence lipid anabolism to modulate macrophage response". In: *PLoS Pathogens* (2009).

[455] Singh, R. et al. "Polyphosphate deficiency in *Mycobacterium tuberculosis* is associated with enhanced drug susceptibility and impaired growth in guinea pigs". In: *Journal of Bacteriology* (2013).

[456] Sinha, A. et al. "PhoP-PhoP interaction at adjacent PhoP binding sites is influenced by protein phosphorylation". In: *Journal of Bacteriology* (2008).

[457] Slawek, J and Arodź, T. "ENNET: inferring large gene regulatory networks from expression data using gradient boosting". In: *BMC Systems Biology* (2013).

[458] Sloggett, C., Goonasekera, N., and Afgan, E. "BioBlend: Automating pipeline analyses within Galaxy and CloudMan". In: *Bioinformatics* (2013).

[459] Smith, J. et al. "Evidence for pore formation in host cell membranes by ESX-1-secreted ESAT-6 and its role in *Mycobacterium marinum* escape from the vacuole". In: *Infection and Immunity* (2008).

[460] Smith, L. J. et al. "Europe PMC Funders Group *Mycobacterium tuberculosis* WhiB1 is an essential DNA-binding protein with a nitric oxide sensitive iron-sulphur cluster". In: *Biochemistry* (2010).

[461] Smollett, K. L., Dawson, L. F., and Davis, E. O. "SigG Does Not Control Gene Expression in Response to DNA Damage in *Mycobacterium tuberculosis H37Rv*". In: *Journal of Bacteriology* (2011).

[462] Smollett, K. L. et al. "Global Analysis of the Regulon of the Transcriptional Repressor LexA, a Key Component of SOS Response in Mycobacterium tuberculosis". In: *Journal of Biological Chemistry* (2012).

[463] Smoot, M. E. et al. "Cytoscape 2.8: new features for data integration and network visualization". In: *Bioinformatics* (2011).

[464] Smyth, G. "Limma: linear models for microarray data". In: *Bioinformatics and Computational Biology Solutions Using R and Bioconductor*. 2005.

[465] Snipen, L., Almoy, T., and Ussery, D. W. "Microbial comparative pan-genomics using binomial mixture models". In: *BMC Genomics* (2009).

[466] Snipen, L. and Liland, K. H. "Micropan: An R-package for microbial pan-genomics". In: *BMC Bioinformatics* (2015).

[467] Snipen, L.-G. and Ussery, D. W. "A domain sequence approach to pangenomics: applications to *Escherichia coli*". In: *F1000Research* (2012).

[468] Solbrig, H. and Prud'hommeaux, E. *Shape Expressions 1.0 Definition*. 2014. URL: `http://www.w3.org/Submission/2014/SUBM-shex-defn-20140602/`.

[469] Song, T. et al. "Critical role of a single position in the -35 element for promoter recognition by *Mycobacterium tuberculosis SigE* and *SigH*". In: *Journal of Bacteriology* (2008).

[470] Splendiani, A. et al. *Semscape Visualizes Semantic Data Landscapes*. 2012. URL: `http://apps.cytoscape.org/apps/semscape`.

[471] Sporny, M., Kellogg, G., and Lanthaler, M. *JSON-LD 1.0 - A JSON-based Serialization for Linked Data*. 2013. URL: `http://www.w3.org/TR/2013/CR-json-ld-20130910/`.

[472] Stanke, M. and Morgenstern, B. "AUGUSTUS: a web server for gene prediction in eukaryotes that allows user-defined constraints". In: *Nucleic Acids Research* (2005).

[473] Stapleton, M. R. et al. “*Mycobacterium tuberculosis* WhiB1 represses transcription of the essential chaperonin GroEL2”. In: *Tuberculosis* (2012).

[474] Stepanauskas, R. “Single cell genomics: An individual look at microbes”. In: *Current Opinion in Microbiology* (2012).

[475] Stewart, A. C., Osborne, B., and Read, T. D. “DIYA: a bacterial annotation pipeline for any genomics lab”. In: *Bioinformatics* (2009).

[476] Stewart, G. R. et al. “Dissection of the heat-shock response in *Mycobacterium tuberculosis* using mutants and microarrays”. In: *Microbiology* (2002).

[477] Stolovitzky, G, Prill, R. J., and Califano, A. “Lessons from the DREAM2 Challenges”. In: *Annals of the New York Academy of Sciences* (2009).

[478] Su, G. et al. “Biological Network Exploration with Cytoscape 3”. In: *Current Protocols in Bioinformatics* (2014).

[479] Sun, H. et al. “Reduced thrombin generation increases host susceptibility to group A streptococcal infection”. In: *Blood* (2009).

[480] Supek, F et al. “A yeast manganese transporter related to the macrophage protein involved in conferring resistance to mycobacteria”. In: *Proceedings of the National Academy of Sciences of the United States of America of the United States of America* (1996).

[481] Sureka, K. et al. “Polyphosphate kinase is involved in stress-induced mprAB-sigE-rel signalling in mycobacteria”. In: *Molecular Microbiology* (2007).

[482] Suzuki, R. and Shimodaira, H. “Pvclust: an R package for assessing the uncertainty in hierarchical clustering”. In: *Bioinformatics* (2006).

[483] Szklarczyk, D et al. “The STRING database in 2011: functional interaction networks of proteins, globally integrated and scored”. In: *Nucleic Acids Research* (2011).

[484] Takao, K. and Miyakawa, T. “Genomic responses in mouse models greatly mimic human inflammatory diseases”. In: *Proceedings of the National Academy of Sciences of the United States of America of the United States of America* (2015).

[485] Tan, J. et al. “ADAGE-Based Integration of Publicly Available Pseudomonas aeruginosa Gene Expression Data with Denoising Autoencoders Illuminates Microbe-Host Interactions”. In: *mSystems* (2016).

[486] Tan, M. P. et al. “Nitrate respiration protects hypoxic *Mycobacterium tuberculosis* against acid- and reactive nitrogen species stresses”. In: *PLoS ONE* (2010).

[487] Taniguchi, Y. et al. “Quantifying *E. coli* Proteome and Transcriptome with Single-Molecule Sensitivity in Single Cells”. In: *Science* (2010).

[488] Tartir, S and Budak Arpinar, I. “Ontology Evaluation and Ranking using OntoQA”. In: *International Conference on Semantic Computing* (2007).

[489] Tatusov, R. L., Koonin, E. V., and Lipman, D. J. “A genomic perspective on protein families”. In: *Science* (1997).

[490] Tatusova, T. et al. “NCBI prokaryotic genome annotation pipeline”. In: *Nucleic Acids Research* (2016).

[491] Tettelin, H. et al. “Comparative genomics: the bacterial pan-genome”. In: *Current Opinion in Microbiology* (2008).

[492] The UniProt Consortium. “UniProt: a hub for protein information”. In: *Nucleic Acids Research* (2015).

[493] Thi, E. P., Lambertz, U., and Reiner, N. E. “Sleeping with the enemy: how intracellular pathogens cope with a macrophage lifestyle”. In: *PLoS Pathogens* (2012).

[494] Thiele, I. and Palsson, B. “A protocol for generating a high-quality genome-scale metabolic reconstruction”. In: *Nature Protocols* (2010).

[495] Thiele, I., Vlassis, N., and Fleming, R. M. T. “fastGapFill: efficient gap filling in metabolic networks”. In: *Bioinformatics* (2014).

[496] Thiele, I. et al. "A community effort towards a knowledge-base and mathematical model of the human pathogen *Salmonella Typhimurium LT2*". In: *BMC Systems Biology* (2011).

[497] Thomas-Chollier, M et al. "RSAT: regulatory sequence analysis tools". In: *Nucleic Acids Research* (2008).

[498] Timmis, K. N. "Pseudomonas putida: a cosmopolitan opportunist par excellence". In: *Environmental Microbiology* (2002).

[499] Tiwari, B. M. et al. "The *Mycobacterium tuberculosis* PE proteins Rv0285 and Rv1386 modulate innate immunity and mediate bacillary survival in macrophages". In: *PLoS ONE* (2012).

[500] Trauner, A. et al. "The dormancy regulator DosR controls ribosome stability in hypoxic mycobacteria". In: *The Journal of Biological Chemistry* (2012).

[501] Tripp, H. J. et al. "Toward a standard in structural genome annotation for prokaryotes". In: *Standards in Genomic Sciences* (2015).

[502] Troudt, J. et al. "*Mycobacterium tuberculosis* sigE mutant ST28 used as a vaccine induces protective immunity in the guinea pig model". In: *Tuberculosis* (2017).

[503] Tufariello, J. M. et al. "Separable roles for *Mycobacterium tuberculosis* ESX-3 effectors in iron acquisition and virulence". In: *Proceedings of the National Academy of Sciences of the United States of America of the United States of America* (2016).

[504] Tullius, M. V. et al. "Discovery and characterization of a unique mycobacterial heme acquisition system". In: *PNAS* (2011).

[505] Tyagi, P. et al. "*Mycobacterium tuberculosis* has diminished capacity to counteract redox stress induced by elevated levels of endogenous superoxide". In: *Free Radical Biology and Medicine* (2015).

[506] Vaas, L. A. I. et al. "opm: an R package for analysing OmniLog(R) phenotype microarray data". In: *Bioinformatics* (2013).

[507] Vallenet, D et al. "MicroScope–an integrated microbial resource for the curation and comparative analysis of genomic and metabolic data". In: *Nucleic Acids Research* (2012).

[508] Van Domselaar, G. H. et al. "BASys: a web server for automated bacterial genome annotation". In: *Nucleic Acids Research* (2005).

[509] VanderVen, B. C. et al. "Novel Inhibitors of Cholesterol Degradation in *Mycobacterium tuberculosis* Reveal How the Bacterium's Metabolism Is Constrained by the Intracellular Environment". In: *PLoS Pathogens* (2015).

[510] Veatch, A. V. and Kaushal, D. "Opening Pandora's Box: Mechanisms of *Mycobacterium tuberculosis* Resuscitation". In: *Trends in Microbiology* (2017).

[511] Vehkala, M. et al. "Novel R pipeline for analyzing biolog phenotypic microarray data". In: *PLoS ONE* (2015).

[512] Veiga, D. F. T., Dutta, B., and Balázsi, G. "Network inference and network response identification: moving genome-scale data to the next level of biological discovery". In: *Molecular Biosystems* (2010).

[513] Via, L. E., Dusanka Deretic, Roseann J. Ulmer, Nina S. Hibler, Lukas A. Huber, V. D. "Arrest of Mycobacterial Phagosome Maturation Is Caused by a Block in Vesicle Fusion between Stages Controlled by rab5 and rab7". In: *Journal of Biological Chemistry* (1997).

[514] Vilchèze, C. et al. "*Mycobacterium tuberculosis* is extraordinarily sensitive to killing by a vitamin C-induced Fenton reaction". In: *Nature Communications* (2013).

[515] W3C organisation. *RDF 1.1 Concepts and Abstract Syntax*. 2014. URL: `http://www.w3.org/TR/rdf11-concepts/`.

[516] Wagner, D. et al. "Elemental Analysis of *Mycobacterium avium*, *Mycobacterium tuberculosis*, and *Mycobacterium smegmatis* Containing Phagosomes Indicates Pathogen-Induced Microenvironments within the Host Cell's Endosomal System". In: *Journal of Immunology* (2013).

[517] Wallace, D. P. *Knowledge management: historical and cross-disciplinary themes*. 2007.

[518] Wang, H., La Russa, M., and Qi, L. S. "CRISPR/Cas9 in Genome Editing and Beyond". In: *Annual Review of Biochemistry* (2016).

[519] Wang, Y et al. "ClpR Protein-like Regulator Specifically Recognizes RecA Protein-independent Promoter Motif and Broadly Regulates Expression of DNA Damage-inducible Genes in Mycobacteria". In: *Journal of Biological Chemistry* (2011).

[520] Warner, D. F. et al. "Essential roles for imuA'- and imuB-encoded accessory factors in DnaE2-dependent mutagenesis in *Mycobacterium tuberculosis*". In: *Proceedings of the National Academy of Sciences of the United States of America of the United States of America* (2010).

[521] Weaver, D. S. et al. "A genome-scale metabolic flux model of *Escherichia coli K-12* derived from the *EcoCyc* database". In: *BMC Systems Biology* (2014).

[522] Weber, T. et al. "AntiSMASH 3.0-A comprehensive resource for the genome mining of biosynthetic gene clusters". In: *Nucleic Acids Research* (2015).

[523] White, M. J. et al. "PepD participates in the mycobacterial stress response mediated through MprAB and SigE". In: *Journal of Bacteriology* (2010).

[524] Wilkinson, M. D. et al. "The FAIR Guiding Principles for Scientific Data management and stewardship". In: *Scientific Data* (2016).

[525] Winden, V. J.C. V. et al. "Mycosins Are Required for the Stabilization of the ESX-1 and ESX-5". In: *mBio* (2016).

[526] Wong, D. et al. "*Mycobacterium tuberculosis* protein tyrosine phosphatase (PtpA) excludes host vacuolar-H+-ATPase to inhibit phagosome acidification". In: *PNAS* (2011).

[527] Wu, X. et al. "Comparative genomics and functional analysis of niche-specific adaptation in Pseudomonas putida". In: *FEMS Microbiology Reviews* (2011).

[528] Xavier, J. C., Patil, K. R., and Rocha, I. "Integration of Biomass Formulations of Genome-Scale Metabolic Models with Experimental Data Reveals Universally Essential Cofactors in Prokaryotes". In: *Metabolic Engineering* (2017).

[529] Yamini M. Ohol, David H. Goetz, Kaman Chan, Michael U. Shiloh, Charles S. Craik, J. S. C. "*Mycobacterium tuberculosis* MycP1 protease plays a dual role in regulation of ESX-1 secretion and virulence". In: *Cell Host & Microbe* (2011).

[530] Yang, M. et al. "Characterization of the interaction between a SirR family transcriptional factor of *Mycobacterium tuberculosis*, encoded by Rv2788, and a pair of toxin-antitoxin proteins RelJ/K, encoded by Rv3357 and Rv3358". In: *FEBS Journal* (2014).

[531] Yang, S., Doolittle, R. F., and Bourne, P. E. "Phylogeny determined by protein domain content". In: *Proceedings of the National Academy of Sciences of the United States of America of the United States of America* (2005).

[532] Yang, X. et al. "Analysis of pan-genome to identify the core genes and essential genes of Brucella spp". In: *Molecular Genetics and Genomics* (2016).

[533] Yeats, C., Bentley, S., and Bateman, A. "New knowledge from old: In silico discovery of novel protein domains in *Streptomyces coelicolor*". In: *BMC Microbiology* (2003).

[534] Yilmaz, P et al. "Minimum information about a marker gene sequence (MIMARKS) and minimum information about any (x) sequence (MIxS) specifications". In: *Nature Biotechnology* (2011).

[535] Yip, A. M. and Horvath, S. "Gene network interconnectedness and the generalized topological overlap measure". In: *BMC Bioinformatics* (2007).

[536] Yu, J et al. "Advances to Bayesian network inference for generating causal networks from observational biological data". In: *Bioinformatics* (2004).

[537] Yus, E. et al. "Impact of genome reduction on bacterial metabolism and its regulation". In: *Science* (2009).

[538] Zhang, B and Horvath, S. "A general framework for weighted gene co-expression network analysis". In: *Statistical Applications in Genetics and Molecular Biology* (2005).

[539] Zhang, B et al. "DDN: a caBIG® analytical tool for differential network analysis". In: *Bioinformatics* (2011).

[540] Zhang, Y., Sun, F., and Yang, H. "CRP Acts as a Transcriptional Repressor of the YPO1635-phoPQ-YPO1632 Operon in *Yersinia pestis*". In: *Current Microbiology* (2015).

[541] Zhang, Y. et al. "Autoregulation of PhoP/PhoQ and positive regulation of the cyclic AMP receptor protein-cyclic AMP complex by PhoP in *Yersinia pestis*". In: *Journal of Bacteriology* (2013).

[542] Zheng, F., Long, Q., and Xie, J. "The function and regulatory network of WhiB and WhiB-like protein from comparative genomics and systems biology perspectives". In: *Cell Biochemistry and Biophysics* (2012).

[543] Zimmermann, M. et al. "Integration of Metabolomics and Transcriptomics Reveals a Complex Diet of *Mycobacterium tuberculosis* during Early Macrophage Infection". In: *mSystems* (2017).

[544] Zomorrodi, A. R. and Maranas, C. D. "Improving the iMM904 S. cerevisiae metabolic model using essentiality and synthetic lethality data". In: *BMC Systems Biology* (2010).

[545] Zomorrodi, A. R. et al. "Mathematical optimization applications in metabolic networks". In: *Metabolic Engineering* (2012).

[546] Zondervan, N. et al. "Regulation of Three Virulence Strategies of *Mycobacterium tuberculosis*: A Success Story". In: *International Journal of Molecular Sciences* (2018).

List of publications

Sipko van Dam, Rui Cordeiro, Thomas Craig, **Jesse CJ van Dam**, Shona H Wood, and João Pedro de Magalhães. "GeneFriends: an online co-expression analysis tool to identify novel gene targets for aging and complex diseases". In: *BMC genomics* 13(1) 2012.

Andrew Gibson, **Jesse CJ Van Dam**, Erik A Schultes, Marco Roos, and Barend Mons. "Towards Computational Evaluation of Evidence for Scientific Assertions with Nanopublications and Cardinal Assertions". In: *Proceedings of the 5th International Workshop on Semantic Web Applications and Tools for Life Sciences (SWAT4LS) Paris, France* 28-30 2012.

Jesse CJ van Dam, Peter J Schaap, Vitor AP Martins dos Santos, and María Suárez-Diez. "Integration of heterogeneous molecular networks to unravel gene-regulation in *Mycobacterium tuberculosis*". In: *BMC Systems biology* 8(1) 2014.

Jesse CJ van Dam, Peter J Schaap, Vitor AP Martins dos Santos, and María Suárez-Diez. "RDF2Graph a tool to recover, understand and validate the ontology of an RDF resource". In: *Journal of biomedical semantics* 6(1) 2015.

Jasper J Koehorst*, **Jesse CJ Van Dam***, Ruben GA Van Heck, Edoardo Saccenti, Vitor AP Martins Dos Santos, María Suárez-Diez, and Peter J Schaap. "Comparison of 432 Pseudomonas strains through integration of genomic, functional, metabolic and expression data". In: *Scientific reports* 6 2016.

*Equal contributions

Jesse CJ van Dam, Jasper J Koehorst, Jon O Vik, Peter J Schaap, and María Suárez-Diez. "Interoperable genome annotation with GBOL, an extendable infrastructure for functional data mining". In: *bioRxiv* 184747 2017.

Jasper J Koehorst,**Jesse CJ van Dam**, Edoardo Saccenti, Vitor AP Martins dos Santos, María Suárez-Diez and Peter J Schaap. SAPP: functional genome annotation and analysis through a semantic framework using FAIR principles. In: *Bioinformatics* 34(8) 2017.

Niels Zondervan, **Jesse CJ van Dam**, Peter J Schaap, Vitor AP Martins dos Santos, and María Suárez-Diez. "Regulation of Three Virulence Strategies of *Mycobacterium tuberculosis*: A Success Story". In: *International Journal of Molecular Sciences* 19(2) 2018.

Erno Lindfors*, **Jesse C.J. van Dam***, Carolyn Ming Chi Lam, Niels Zondervan, Vitor A.P. Martins dos Santos, and Maria Suarez-Diez. "SyNDI: Synchronous Network Data Integration framework". In: *BMC Bioinformatics* 19 2018.

*Equal contributions

Nhung Pham, Ruben Van Heck, **Jesse C.J. van Dam**, Peter J Schaap, Edoardo Saccenti, María Suárez-Diez. "Consistency, Inconsistency and Ambiguity of Metabolite Names in Biochemical Databases Used for Genome Scale Metabolic Modelling". In: *bioRxiv* 503664 2018.

Jesse C.J. van Dam, Vitor A.P. Martins dos Santos, Peter J. Schaap and Maria Suarez-Diez. Bio-Growmatch: "High quality automatic model building through automated incorporation of phenotype data". In preparation

Jesse C.J. van Dam, Jasper J. Koehorst, Peter J. Schaap, and Maria Suarez-Diez. The Empusa code generator: bridging the gap between the intended and the actual content of RDF resources. In preparation

Gerben D. A. Hermes, Poncheewin W.,**J. C.J. van Dam**, Jasper J Koehorst, Maria Suarez-Diez, Peter J. Schaap, and Smidt Hauke. High-throughput (16S rRNA) amplicon analysis through semantic framework using FAIR principles. In preparation

Overview of completed training activities

Discipline specific activities	**Year**	**Country**
SWAT4LS conference	2012	France
NCSB conference	2012	Netherlands
Microme course	2012	France
Pattern recognition course	2013	Netherlands
SBNL symposium	2013	Netherlands
SWAT4LS conference	2013	United Kingdom
SBNL symposium	2014	Netherlands
NL@SB conference	2014	Netherlands
SWAT4LS conference	2014	Germany
Protein structures: production, prowess, power, promises, and problems course	2015	Netherlands
SWAT4LS conference + presented poster	2015	United Kingdom
General courses		
VLAG PhD week	2012	Netherlands
Project and time management	2013	Netherlands
Techniques for writing and presenting a scientific Paper	2013	Netherlands
Entrepreneurship in- and outside science	2014	Netherlands
Writing grant proposals	2015	Netherlands
Speed reading	2015	Netherlands
Optional activities		
Preparation of research proposal course	2012	Netherlands
Weekly group meetings	2012-2017	Netherlands
BWise seminar series	2014-2017	Netherlands
SSB retreat	2014	Netherlands
SSB retreat	2015	Netherlands
PhD trip	2015	United States of America

Summary

The goal of this thesis is to improve the prediction of genotype to phenotype associations with a focus on metabolic phenotypes of prokaryotes. This goal is achieved through data integration, which in turn required the development of supporting solutions based on semantic web technologies. **Chapter 1** provides an introduction to the challenges associated to data integration. Semantic web technologies provide solutions to some of these challenges and the basics of these technologies are explained in the Introduction. Furthermore, the basics of constraint based metabolic modeling and construction of genome scale models (GEM) are also provided. The chapters in the thesis are separated in three related topics: **chapters 2, 3** and **4** focus on data integration based on heterogeneous networks and their application to the human pathogen *M. tuberculosis*; **chapters 5, 6, 7, 8** and **9** focus on the semantic web based solutions to genome annotation and applications thereof; and **chapter 10** focus on the final goal to associate genotypes to phenotypes using GEMs.

Chapter 2 provides the prototype of a workflow to efficiently analyze information generated by different inference and prediction methods. This method relies on providing the user the means to simultaneously visualize and analyze the coexisting networks generated by different algorithms, heterogeneous data sets, and a suite of analysis tools. As a show case, we have analyzed the gene co-expression networks of *M. tuberculosis* generated using over 600 expression experiments. Hereby we gained new knowledge about the regulation of the DNA repair, dormancy, iron uptake and zinc uptake systems. Furthermore, it enabled us to develop a pipeline to integrate ChIP-seq dat and a tool to uncover multiple regulatory layers.

In **chapter 3** the prototype presented in **chapter 2** is further developed into the Synchronous Network Data Integration (SyNDI) framework, which is based on Cytoscape and Galaxy. The functionality and usability of the framework is highlighted with three biological examples. We analyzed the distinct connectivity of plasma metabolites in networks associated with high or low latent cardiovascular disease risk. We obtained deeper insights from a few similar inflammatory response pathways in *Staphylococcus aureus* infection common to human and mouse. We identified not yet reported regulatory motifs associated with transcriptional adaptations of *M. tuberculosis*.

In **chapter 4** we present a review providing a systems level overview of the molecular and cellular components involved in divalent metal homeostasis and their role in regulating the three main virulence strategies of *M. tuberculosis*: immune modulation, dormancy and phagosome escape. With the use of the tools presented in **chapter 2** and **3** we identified a single regulatory

cascade for these three virulence strategies that respond to limited availability of divalent metals in the phagosome.

The tools presented in **chapter 2** and **3** achieve data integration through the use of multiple similarity, coexistence, coexpression and interaction gene and protein networks. However, the presented tools cannot store additional (genome) annotations. Therefore, we applied semantic web technologies to store and integrate heterogeneous annotation data sets. An increasing number of widely used biological resources are already available in the RDF data model. There are however, no tools available that provide structural overviews of these resources. Such structural overviews are essential to efficiently query these resources and to assess their structural integrity and design. Therefore, in **chapter 5**, I present RDF2Graph, a tool that automatically recovers the structure of an RDF resource. The generated overview enables users to create complex queries on these resources and to structurally validate newly created resources.

Direct functional comparison support genotype to phenotype predictions. A prerequisite for a direct functional comparison is consistent annotation of the genetic elements with evidence statements. However, the standard structured formats used by the public sequence databases to present genome annotations provide limited support for data mining, hampering comparative analyses at large scale. To enable interoperability of genome annotations for data mining application, we have developed the Genome Biology Ontology Language (GBOL) and associated infrastructure (GBOL stack), which is presented in **chapter 6**. GBOL is provenance aware and thus provides a consistent representation of functional genome annotations linked to the provenance. The provenance of a genome annotation describes the contextual details and derivation history of the process that resulted in the annotation. GBOL is modular in design, extensible and linked to existing ontologies. The GBOL stack of supporting tools enforces consistency within and between the GBOL definitions in the ontology.

Based on GBOL, we developed the genome annotation pipeline SAPP (Semantic Annotation Platform with Provenance) presented in **chapter 7**. SAPP automatically predicts, tracks and stores structural and functional annotations and associated dataset- and element-wise provenance in a Linked Data format, thereby enabling information mining and retrieval with Semantic Web technologies. This greatly reduces the administrative burden of handling multiple analysis tools and versions thereof and facilitates multi-level large scale comparative analysis. In turn this can be used to make genotype to phenotype predictions.

The development of GBOL and SAPP was done simultaneously. During the development we realized that we had to constantly validated the data exported to RDF to ensure coherence with the ontology. This was an extremely time consuming process and prone to error, therefore we developed the Empusa code generator. Empusa is presented in **chapter 8**.

SAPP has been successfully used to annotate 432 sequenced *Pseudomonas* strains and integrate the resulting annotation in a large scale functional comparison using protein domains. This comparison is presented in **chapter 9**. Additionally, data from six metabolic models, nearly a thousand transcriptome measurements and four large scale transposon mutagenesis experiments were integrated with the genome annotations. In this way, we linked gene essentiality, persistence and expression variability. This gave us insight into the diversity, versatility and evolutionary history of the *Pseudomonas* genus, which contains some important pathogens as well some useful species for bioengineering and bioremediation purposes.

Genome annotation can be used to create GEM, which can be used to better link genotypes to phenotypes. Bio-Growmatch, presented in **chapter 10**, is tool that can automatically suggest modification to improve a GEM based on phenotype data. Thereby integrating growth data into the complete process of modelling the metabolism of an organism.

Chapter 11 presents a general discussion on how the chapters contributed the central goal. After which I discuss provenance requirements for data reuse and integration. I further discuss how this can be used to further improve knowledge generation. The acquired knowledge could, in turn, be used to design new experiments. The principles of the dry-lab cycle and how semantic technologies can contribute to establish these cycles are discussed in **chapter 11**. Finally a discussion is presented on how to apply these principles to improve the creation and usability of GEM's.

Acknowledgements

Finally, after all those years of hard work with all the great support I had, I managed to finish my thesis with prided. I am very thankful for the opportunity to work during my PhD with great freedom. I am thankful for the support given by all colleagues, collaborators and especially my wife Corien. Corien has given me a good family life, despite the demanding task off raising and giving birth to my three beloved darlings. In 2013 my son Alvar was born, in 2015 my daughter Mathilde and in 2017 my son Wederik. I am very thankful to my wife Corien for taking care of three children and always supporting me and giving me the freedom to also spend time with my friends and spend time away on trips needed for my PHD.

Jasper, I am grateful for having you as my friend and partner in crime in the development of the semantic annotation pipeline and associated tools. I was mostly busy with the technical parts of semantic web technologies, whereas Jasper concentrated more on the biological application of it. Without your support it would never become such a great application as it has become now. You always motivated me by showing the value of the work we did. Together with Niels we had great discussions while walking outside.

Maria, thank you for always providing with great support as you helped, teached and motivated me despite my dyslexia, allowing me to write good scientific papers. Despite myself being sometimes pessimistic about my own work, you gave me the motivation to keep going and with great results at the end. Maria, you have great knowledge about nearly anything and always know how to efficiently finish of the last missing details.

Peter, you always came with ideas, asking me difficult questions which allowed me to focus on the important elements, which definitely helped in the final result. You always gave me good support when work and my family live where biting each other. You also allowed me to act with great freedom within the group. With great proud together with Jasper, Maria, Peter and Benoit we created the basis for another semantic web group in the Netherlands.

With Niels, a great friend of mine, I always had good discussions, about the personal and family life, software development, metabolism and *Mycobacterium Tuberculosis*. Together, while using my tool Diva and by reading a lot of literature (>300 papers) we created a cool map about the infection machinery of the complex bug Mycobacterium tuberculosis.

Jolanda, a good friend of mine, gave me the support needed when I was in the middle of my thesis. At the times it became more difficult and concentration was dwindling, you supported me and gave me a secure feeling allowing me to reorganize and finish my thesis. I am very thankful for your support.

Together with Maria, there was Eduardo, who was always there to discuss any statistical questions. You talk with great passion about statistics and was always willing to help solve statistical problems.

Vitor was my professor of the group, I would like to thank him for offering me the PHD position within his group and for being the person who raised the funds for the group. Vitor together with Peter helped me initiate some of the good collaborations, which I needed to finish my Thesis.

I would like to thank Vitor also for all the trips that I could make, which where needed for the collaborations. Despite being away from the family, I always had great fun with my travel companions, which included Jasper, Peter, Ruben and Mark.

In my second year of my PHD thesis, we had a great group of colloquies, which included Benoit, Maarten, Bastian, Nicolas, Jasper, Niels, Dorett, Ruben, Michiel and Milad with I had lot of fun, carting, laser gaming and paint balling. Within this group we formed a Raspberry pi train group. Every week we had an evening in which we tried to develop a robot-based game in which we had a lot of hacking fun, while eating pizza's. I still think back with great pleasure to this time, as it was a welcome motivator to work in the group.

Ruben, was our smart metabolic model engineer, who did a some great work and was always willing to help. I am thankful for all the fun he organized for us and for all the useful discussions we had about the metabolic modeling.

I also would like to thank Nhung, who followed up Ruben as metabolic model engineer. Together with Ruben we worked on the metabolic name space issue. Nhung was always friendly and I enjoyed working together with you.

Dorett, was a great bioinformatician with a strong biological insight about metabolism. Together with Michel, Ruben and others in the Pi train group we always had great fun. I always enjoyed the conversations we had together.

Michel was like me a fan of coding software, we had some good fun in the creation of tool for metabolic engineering, which was sadly enough never followed up.

Benoit brought some crazy fun to the group and was always in for some fun. If I needed a new gadget or had some funny computer problems, then Benoit was the go to person. He was always great fan of the semantic web technologies and we had some useful discussions about it.

Bart was our IT specialist who maintained and kept our servers running. Thanks to him I was able to use the servers without the need to do any maintenance, which saved me a lot of time.

For all my computer, network and general infrastructure issues there was always the great support from Wim, as he was always there ready to help you with pleasure.

I would like to thank Maarten and Ruben for all the work they did to organize the PHD trip, for SSB. The trip was great experience and well organized. I will never forget we just made the transfer as a group after we got broken up by the border security.

I would like to thank Mark, for getting me interested in the metabolic modeling and for the support he gave to me when I was still a master student. Together we made the trip to Eytan Ruppin in Tel Aviv with great pleasure. Like me, Mark was doing a lot of construction works in his home which was fun to discus.

Furthermore, I would like to thank Rienk, who also helped me to understand the metabolism of Mycobacterium Tuberculosis. He was responsible for the important discovery of the cholesterol degradation pathway in Mycobacterium tuberculosis.

Bastian was for long time one of my office mates, I had some good discussions about secondary metabolism and he always brought that extra bit of fun in all the group activities we had within the group.

I would like to thank Nikolas for using our tools and helping me and Jasper improving our tools from a user point of view.

Rob was a great organizer, effectively organizing the Theoretical Systems and Synthetic Biology Group. Besides that, he did a lot to organize the IGEM projects, which resulted in the presence of many nice students within the group, including Lisa, which became a friend of mine. I am very thankful for the work he did to organizing the IGEM projects.

I would like to thank Lisa, Wiebe, Nicole, Mara and Linde for playing D&D together and I would like to thank Guus for organizing it. We had great fun together killing the monsters.

I would like to thank Ruben, Dorett, Wen and Margo for the climbing evenings we had in Utrecht, Amersfoort and Nijmegen.

I would like to thank Javier, Wiebe, Gerben, Peter, Jasper, Wasim, Hauke and Detmar,Clara for helping with the creation of the NG-Tax tool, which has a great use within the microbiology group.

I would like to thank Erno Lindfors for the work he did to develop Syndi tool.

Thanks to the partners in the Microme project I was able to make chapter 10 of my thesis. During the meetings, which were at some great locations, we had good discussions and hackathons. I am thankful for the good collaboration with all the members within the project.

I would like to thank my partners in the digital salmon project, for all the feedback allowing us to further improve GBOL and associated tools. It was always a great pleasure to work with you guys and I enjoyed the rock climbing.

I would like the thank the Marco Roos, Mark Thompson, Erik Schultes, Barend Mons, Eleni Mina, and Rajaram Kaliyaperumal in the Bio-semantics group in Leiden and Nirupama as member in our group, for getting me interested in the semantic web technologies in the first place and for all the subsequent collaboration, support and meetings. I always felt very welcome in your group.

My special thanks to my external collaborator Erik Prud'hommeaux, who was responsible for the development of SPARQL an ShEx. He was a great supporter for the semantic web technologies and we had some fruitful discussions about the technology.

I also would like to thank Andra Waagmeester, with whom I and Jasper had useful discussions and hackathons related to the semantic web technologies applied to metabolism.

At last I am very thankful to my family who supported me. My parents, Wouter and Sipko supported me with time and money to buy a house and by helping me with the construction works in the house. My parents and my mother in law supported us with the difficult time we had around the pregnancies and birth of our darlings. I am also grateful to Sipko who got me interested in the field of biology and bio-informatics, which gave me the opportunity to start my master study and PHD in the first place. I am thankful to my parents, Wouter, Bob and Sipko for always being there as a family for me, giving me a solid base to start with.

The research described in this thesis was financially supported by the “NCSB Stimulus 2013 project: data integratie” project of Netherlands Consortium for Systems Biology, DigiSal project (Project reference: 248792) of the research council of Norway, the European Union Horizon2020 project EmPowerPutida (Project reference: 635536), and the European Union’s 7th Framework projects Microme (Project reference: 222886), INFECT (Project reference: 321529), MycoSynVac(project reference: 634940) and SysteMTB(project reference: 241587). Parts of this work was carried out on the Dutch national e-infrastructure with the support of the SURF foundation.

Cover design by: Jesse van Dam

Printed by: Digiforce, Wageningen