I would suggest we avoid the comparison to sequence databases or genome sequence platforms (e.g. assembly, gene annotation etc). We really don't have the capacity to replace EnteroBase, Genbank, IRIDA, etc. I see Spfy as a repository of E.coli phenotypes important for public health. It allows users to rapidly predict phenotypes and retrieve the phenotypes of historical strains to: track trends, observe context amoung evolutionary related strains, identify biomarkers, etc. In other words, Spfy provides WGS-based phenotypes + big picture comparative genomics. The big picture is only possible by 1. recomputing everything everytime, or 2. storing the results as genomes are analysed. With size of WGS data, option 2 is only viable option Title may need work. Should it mention comparative genomics? Can't be too long thou

Spfy: an integrated graph database for real-time prediction of Escherichia coli phenotypes and downstream comparative analyses

Kevin K Le*[1], Matthew D Whiteside[1], James Hopkins[1], Victor PJ Gannon[1] and Chad R Laing†[1]

[1]National Microbiology Laboratory at Lethbridge, Public Health Agency of Canada, Twp Rd 9-1, Lethbridge, AB, T1J 3Z4, Canada

February 28, 2018

Abstract

Public health laboratories are currently moving to whole-genome sequencing based analyses, and require rapid prediction of relevent pathogen phenotypes impacting health. Current workflows in comparative computational genomics rely on chaining different analysis software together, but lack storage and retrieval methods for the generated results. Reference laboratories do not care about data retrieval, only real-time results. Researchers care more about context and thus would use a historical record of phenotypes for all predicted strains To solve this problem, we have created Spfy, which uses a graph database to store and retrieve results from computational workflows. The newly developed Spfy platform facilitates rapid phenotype identification, as well as the efficient storage and downstream comparative anlaysis of tens of thousands of genome sequences. Though generally applicable to bacterial genome sequences, Spfy currently contains X *Escherichia coli* genomes, for which *in-silico* serotype and Shiga-toxin subtype, as well as the presence of known virulence factors and antimicrobial resistance determinants have been computed. Spfy links the results and metadata to the genome sequences through a standardized ontology, which facilitates hypothesis testing in fields ranging from population genomics to epidemiology, while mitigating the recomputing of analyses. The graph approach is flexible, and can accommodate new analysis software modules as they are developed, and easily link new results to those already stored. Integrated data storage and analyses are currently necessary as the number of publicly available whole genome sequences is currently in the hundreds of thousands, with millions likely to be available within the next few years. should we mention how many ecoli genomes are available? It might prompt reviewers to ask why we don't have them in spfy

Database URL: `https://lfz.corefacility.ca/superphy/spfy/`.

1 Introduction

Whole genome sequencing (WGS) can in theory provide the entire genetic content of an organism. This unparalleled resolution and sensitivity has recently transformed public-health surveillance and outbreak response [1, 2]. Additionally, the identification of novel disease mechanisms [3, 4], and rapid clinical diagnoses and reference lab tests based on the specific mechanism of disease are now possible. [5, 6].

The rapid characterization and comparison of bacterial pathogens relies principally on the combination of outputs from multiple software programs that are targeted for specific applications. Examples include the identification of known antimircrobial resistance (AMR) genes, through software such as the Resistance Gene Identifier (RGI) [7], [8], [9], and ARIBA [10]; or the identification of known virulence factor genes (VF) through software such as VirulenceFinder [8], SRST2 [11], and GeneSippr [12]. For clinical diagnoses and comparisons, individual species can be first divided into subtypes with complementing AMR and VF results.

*kevin.le@canada.ca

†chad.laing@canada.ca

Software methods for subtyping rely on intraspecies genes or genomic regions, and are targeted through software such as Phylotyper [13], SerotypeFinder [14], the EcOH dataset applied through SRST2 [15], and V-Typer [16]. These methods represent *in-silico* analogues of traditional wet-lab tests, which allows new WGS results to be viewed in the context of historical tests, and greatly expedites the analyses of newly sequenced genomes.

Comprehensive platforms that combine individual programs into a cohesive whole also exist. These include free platforms such as the Bacterium Analysis Pipeline (BAP) [17], and the Pathosystems Resource Integration Center (PATRIC) [18]. Commercial applications, such as Bionumerics, which is used by PulseNet International for the analyses of WGS data in outbreak situations also exist, and offer support as well as accredited, standardized tests [19]. These platforms are designed to be applied to individual projects [].

WGS of bacterial pathogens have recently accumulated in public databases in the hundreds of thousands, with millions set to be available within the next few years. For *Escherichia coli* alone, there are over sixty thousand publicly available genomes in EnteroBase `https://enterobase.warwick.ac.uk/` and three million sequenced genomes in GenBank [20]. Many of the comparative analyses that are currently used in the analyses of bacterial genomes are broadly useful, and therefore computed multiple times for the same genomes. An effective method to mitigate the recomputing of analyses, is to make the storage and retrieval of results part of the analyses platform, and effectively linked to the genomes of interest through a standardized ontology. Such measures can help ensure the rapid response times required for public health applications, and allow results to be integrated and progressively updated as new data becomes available. should we mention how many ecoli genomes are available? It might prompt reviewers to ask why we don't have them in spfy

We have previously developed Superphy [21], an online predictive genomics platform targeting *E. coli*. Superphy integrates pre-computed results with domain-specific knowledge to provide real-time exploration of publicly available genomes. While this tool has been useful for the thousands of pre-computed genomes in its database, the current pace of genome sequencing requires real-time predictive genomic analyses of tens-, and soon hundreds-of-thousands of genomes, and the long term storage and referencing of these results, something that the original SuperPhy platform was incapable of.

In this study, we present the Spfy update to the SuperPhy platform, which integrates a graph database with real-time analyses; this integration avoids recomputing identical analyses. Graph-based result storage also allows retrospective comparisons as more genomes are sequenced or populations change, and is flexible, accommodating new analysis modules as they are developed. The database is available at `https://lfz.corefacility.ca/superphy/spfy/`.

2 FUNCTIONALITY

Spfy provides rapid in-silco versions of common reference laboratory tests for the analyses of *E. coli*. It supports the following *in-silico* subtyping options: serotyping, through both O- and H-antigen identification ectyper ref; Shiga-toxin 1 (Stx1), Shiga-toxin 2 (Stx2), and Intimin typing using Phylotyper [13], VF gene determination using ECTyper `https://github.com/phac-nml/ecoli_serotyping`, and AMR annotation using the RGI program [7].

Spfy also performs pangenome analyses using Panseq [22], and provides machine learning modules for biomarker discovery among groups using Scikit-learn [23].

Spfy handles all of the analyses tasks by dividing them into subtasks, which are subsequently distributed

across a built-in task queue. Results are converted into individual graphs and stored within a larger graph database according to the standard ontologies GenEpiO [24], FALDO [25], and TypOn [26], where metadata including genotypes, biomarkers, host, source, and statistical significance testing of genome markers for user-defined groups are stored.

By integrating task distribution with graph storage, Spfy enables large-scale analyses, such as epidemiological association studies. Any data type or relation in the graph is a valid option for analysis. This means that genomes can be compared on the basis of the presence or absence of pan-genome regions, serotype, subtyping data, or provided metadata such as location or host-source.

3 IMPLEMENTATION

The server-side code for Spfy, graph generation, and analysis modules, are developed in Python, with the front-end website developed using the React JavaScript library `https://facebook.github.io/react/`. When new data is added to the database, the following steps are taken:

i) The upload begins through the website, where user-defined analyses options are selected. The results of these analyses are immediately reported to the user following their completion, while all other non-selected analyses are subsequently completed in the background and stored in the database without interaction from the user. The public web service accepts uploads of up to 200 MB (approximately 50 *E. coli* genomes uncompressed, or 120 genomes compressed) at a time, though an unlimited amount of data can be submitted to a local instance.

ii) User-selected analyses are enqueued into the Redis Queue `http://python-rq.org/` task queue. Redis Queue consists of a Redis Database `https://redis.io/` and task queue workers which run as Python processes.

iii) The workers dequeue the analyses, run them in parallel, and temporarily store results in the Redis database.

iv) Python functions parse the results and permanently store them in Blazegraph `https://www.blazegraph.com/`, the graph database used for Superphy.

3.1 Data Storage

Semantic web technologies describe the relationships between data [27], and graph databases for the storage of this information has been proposed as a open standard for sharing public information [28]. For biological data, individual data points can be a genome, contiguous DNA sequence, or gene, and these are linked together in a searchable graph structure using existing ontologies, including annotations from given ontologies. This system is flexible and allows novel data to be incorporated into the existing graph.

The permanent storage of results is as a one-time cost to avoid recomputation when the same analysis is re-run. For analyses, Spfy searches the graph for all data points annotated with the queried ontology term. This graph data is then converted into the required structure, usually numerical arrays, or as required for the given analysis module. In a graph database, a search can begin at any node or attribute. This is in contrast to a SQL database which requires a predefined schema, or a NoSQL database which treats data as documents with varying structure. For example, the addition of a new analysis module would typically require a new table definition in a SQL database, or the addition of a new document type in a NoSQL database. With a graph database, new nodes or attributes are added and then connected to existing data, removing the need for explicit joins or data conversions. Additionally, data can be added to Spfy in parts, and the database will infer the correct connections between the data.

3.2 Web design

The front-end website is written as a single-page application. To ensure a familiar user interface, we followed the Material Design specification `https://material.io/`, published by Google, surrounding a card-based design. (see Figure 2) Both the task selection and result displays follow the same design pattern: while data storage is graph-based, the results of various analysis modules are presented to users in a familiar tabular structure and available for download as .csv spreadsheet files. (see Figure 3)

No account creation is required to use the platform. A sharable token is automatically created for users upon entering the website, and is embedded into the website address. Users can share results by copying their URL, and files submitted from different computers using the same token will be visible to anyone with the same link.

3.3 Service Virtualization

Docker `https://www.docker.com/` is a virtualization technology to simulate self-contained operating systems on the same host computer, without the overhead of full hardware virtualization [29]. The Spfy platform

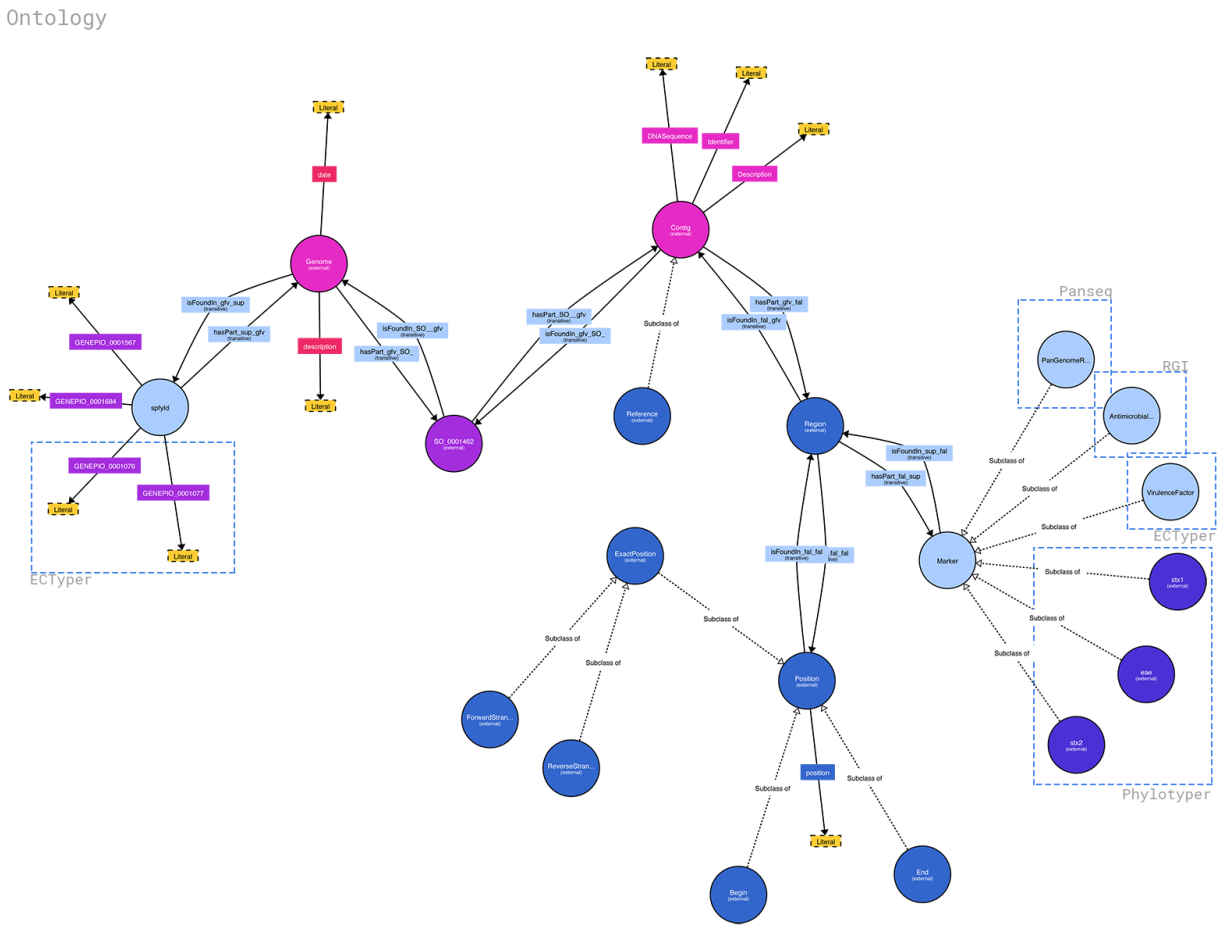


Figure 1: Structure of Spfy's graph database. Brackets highlight the source of different data points and the software it was generated from. These parts are added in as the analyis modules complete, at varying times, and the overall connections are inferred by the database. Non-bracketed sections are sourced from the uploaded genome files or user-supplied metadata.

depends on a series of webservers, databases, and task workers, and uses Docker to compartmentalize these services, which are then networked together using Docker-Compose `https://docs.docker.com/compose/`.

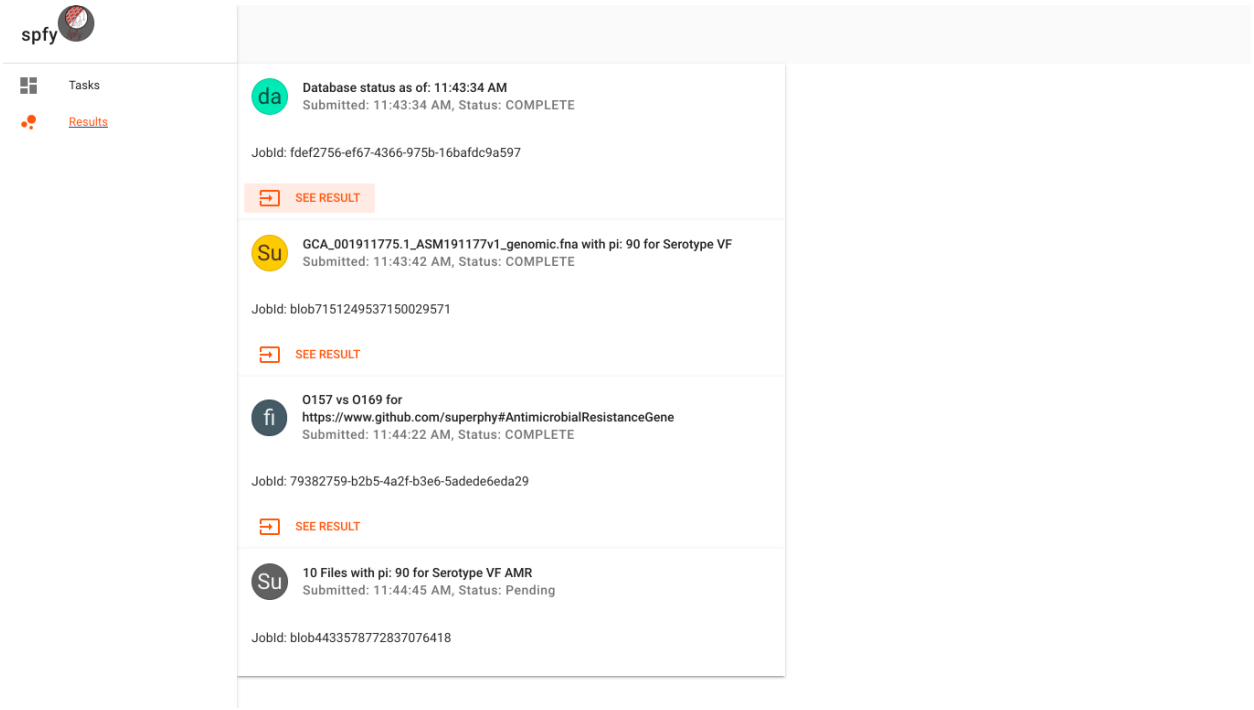
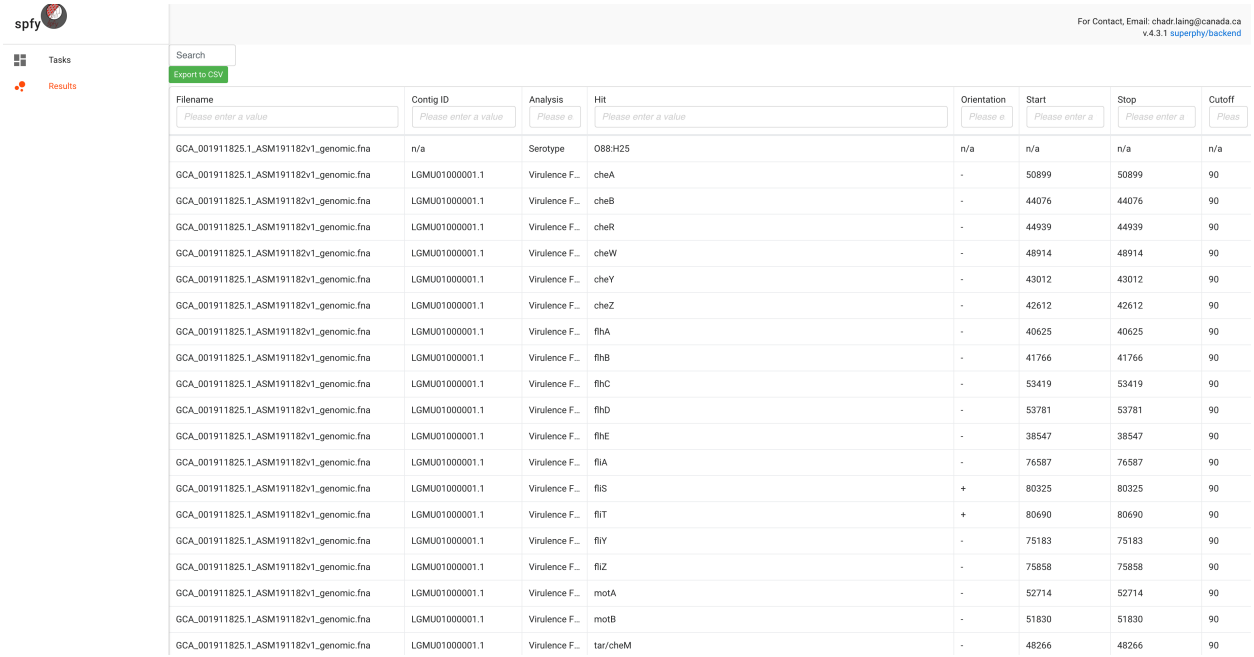


Figure 2: First entry into the results interface for submitted tasks. Cards represent individual tasks and the entire collection can be viewed by sharing the URL with the embedded token.



Filename	Contig ID	Analysis	Hit	Orientation	Start	Stop	Cutoff
GCA_001911825.1_ASM191182v1_genomic.fna	n/a	Serotype	O88:H25	n/a	n/a	n/a	n/a
GCA_001911825.1_ASM191182v1_genomic.fna	LGMU01000001.1	Virulence F...	cheA	-	50899	50899	90
GCA_001911825.1_ASM191182v1_genomic.fna	LGMU01000001.1	Virulence F...	cheB	-	44076	44076	90
GCA_001911825.1_ASM191182v1_genomic.fna	LGMU01000001.1	Virulence F...	cheR	-	44939	44939	90
GCA_001911825.1_ASM191182v1_genomic.fna	LGMU01000001.1	Virulence F...	cheW	-	48914	48914	90
GCA_001911825.1_ASM191182v1_genomic.fna	LGMU01000001.1	Virulence F...	cheY	-	43012	43012	90
GCA_001911825.1_ASM191182v1_genomic.fna	LGMU01000001.1	Virulence F...	cheZ	-	42612	42612	90
GCA_001911825.1_ASM191182v1_genomic.fna	LGMU01000001.1	Virulence F...	flhA	-	40625	40625	90
GCA_001911825.1_ASM191182v1_genomic.fna	LGMU01000001.1	Virulence F...	flhB	-	41766	41766	90
GCA_001911825.1_ASM191182v1_genomic.fna	LGMU01000001.1	Virulence F...	flhC	-	53419	53419	90
GCA_001911825.1_ASM191182v1_genomic.fna	LGMU01000001.1	Virulence F...	flhD	-	53781	53781	90
GCA_001911825.1_ASM191182v1_genomic.fna	LGMU01000001.1	Virulence F...	flhE	-	38547	38547	90
GCA_001911825.1_ASM191182v1_genomic.fna	LGMU01000001.1	Virulence F...	fliA	-	76587	76587	90
GCA_001911825.1_ASM191182v1_genomic.fna	LGMU01000001.1	Virulence F...	fliS	+	80325	80325	90
GCA_001911825.1_ASM191182v1_genomic.fna	LGMU01000001.1	Virulence F...	fliT	+	80690	80690	90
GCA_001911825.1_ASM191182v1_genomic.fna	LGMU01000001.1	Virulence F...	fliY	-	75183	75183	90
GCA_001911825.1_ASM191182v1_genomic.fna	LGMU01000001.1	Virulence F...	fliZ	-	75858	75858	90
GCA_001911825.1_ASM191182v1_genomic.fna	LGMU01000001.1	Virulence F...	motA	-	52714	52714	90
GCA_001911825.1_ASM191182v1_genomic.fna	LGMU01000001.1	Virulence F...	motB	-	51830	51830	90
GCA_001911825.1_ASM191182v1_genomic.fna	LGMU01000001.1	Virulence F...	tar/cheM	-	48266	48266	90

Figure 3: An expanded result for a subtyping task. While data storage in Spfy is graph-based, a familiar tabular structure is presented to users. The result can also be shared using the URL with the embedded token.

(see Figure 4) Docker integration ensures that software dependencies, which are typically manually installed [30, 22, 11, 31], are instead handled automatically.

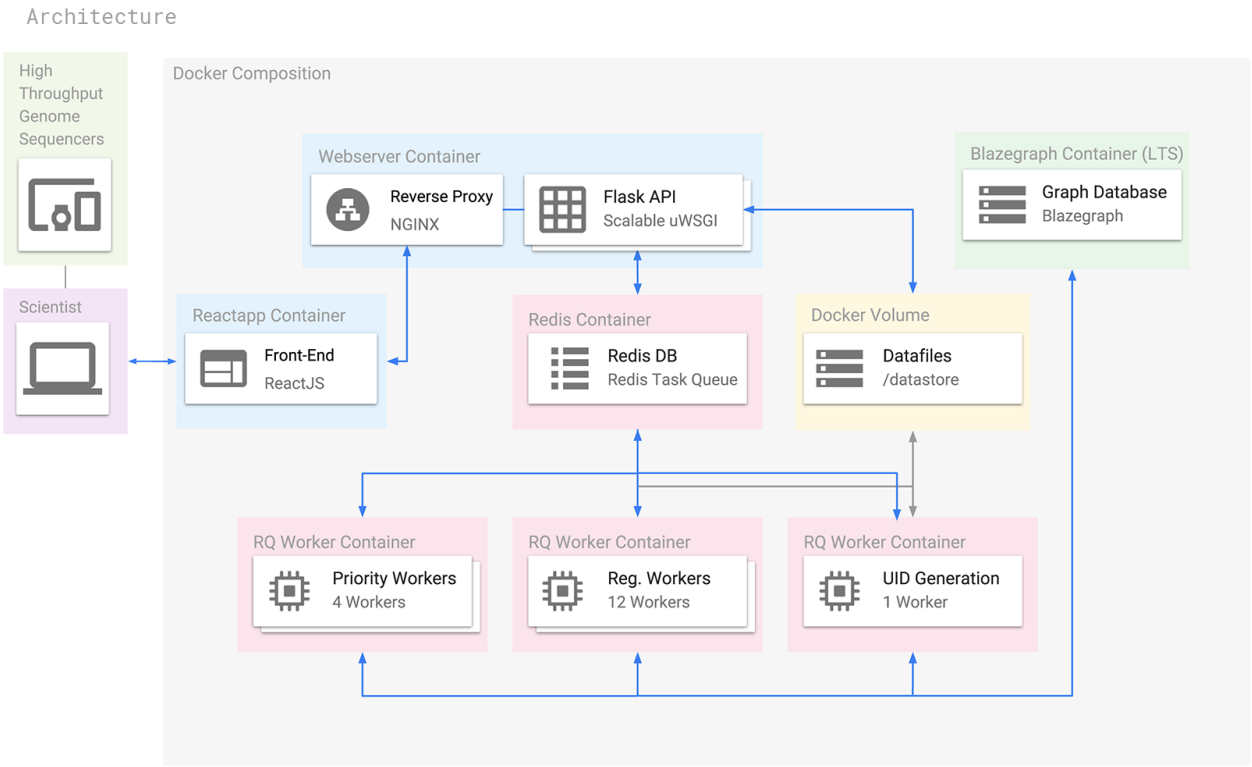


Figure 4: The various Docker containers used in Spfy. Arrows represent the connections between different containers, and the entire platform can be recreated with a single command using Docker-Compose.

One of the key benefits of using common-place technologies is the compatibility with other infrastructure resources. Docker containers are widely supported by cloud computing services: Amazon Web Services (AWS) `https://aws.amazon.com/docker/`, Google Cloud Platform (GCloud) `https://cloud.google.com/container-engine/`, and Microsoft Azure `https://azure.microsoft.com/en-us/services/container-service/`, and self-hosted cloud computing technologies such as OpenStack `https://wiki.openstack.org/wiki/Docker`. Spfy packages compute nodes as reproducible Docker containers, and allows the platform to easily scale to demand.

3.4 Continuous integration

Our tests for functionality and backwards compatibility run on TravisCI `https://travisci.io`, a continuous integration (CI) platform. The individual tests use PyTest `https://doc.pytest.org/`, and the current build status can be checked either on our GitHub repository or at `https://travis-ci.org/superphy/backend`. TravisCI also builds the core Docker images for Spfy, and uploads them to Docker Hub `https://hub.docker.com/u/superphy/`.

4 RESULTS

Spfy was tested with 10,243 public *E. coli* assembled genomes from Enterobase, storing every sequence and the results for all included analysis modules. This included: serotypeing (O-antigen, H-antigen), toxin subtyping (Shiga-toxin 1, Shiga-toxin 2, and Intimin), the identification of VF and AMR determinants, and

determination of pan-genome content *E. coli*. The resulting database had 17,820 nodes and 3,811,473 leaves, with 1,125,909,074 object properties. Spfy has been up since May 2017. The server accepts assembled *E. coli* genomes with the *.fasta* or *.fna* extensions. Submissions are subjected to quality control, ensuring the submitted genomes are *E. coli* sequences before subsequent analyses are run.

Table 1: Database statistics for the various types of nodes, leaves, and associated entries in the graph database, Blazegraph. The database stores all results from every included analysis module and associated metadata. Numbers are representative for analysis of 10,243 assembled genomes.

name	indexType	m	height	nnodes	nleaves	nentries	nodeBytes	leafBytes	totalBytes
_globalRowStore	BTree	32	1	1	6	102	193	8537	8730
kb.lex.BLOBS	BTree	692	2	17	6727	3227686	90596	60187016	89331537662
kb.lex.ID2TERM	BTree	905	2	88	39912	18080577	515355	293436043	2058798455
kb.lex.TERM2ID	BTree	193	3	1153	147978	18080577	5596557	1152764154	1158360711
kb.spo.JUST	BTree	284	3	13213	2042532	299426518	77979362	15527483178	15605462540
kb.spo.OSP	BTree	708	3	1649	639325	262364538	11448125	3760927987	3772376112
kb.spo.POS	BTree	990	2	864	463188	262364538	8347594	2246478879	2254826473
kb.spo.SPO	BTree	1024	2	835	471805	262364538	10308603	2661857997	2672166600

5 DISCUSSION

Many bioinformatics software programs have been developed *ad hoc*, with individual researchers and laboratories developing software specific to their environment [32]. Such tools were often script-based, with custom data formats, and only suitable for small collections of data [32]. Recent efforts [33, 17] have focused on providing a common web interface for these programs, while still returning the same result files. However, many subsets of biology now require the analyses of big-data, where inputs are taken from a variety of analysis programs, and involve large-scale data warehousing [34]. The ability to integrate data from different source technologies, merge submissions from other labs, and distribute computations over fault-tolerant systems are now required for types of analyses that need to be performed [34].

One of the key goals in developing Spfy was to accommodate and store a variety of result formats, and then to make the data from these results retrievable and usable as inputs for downstream analyses, such as predictive biomarker discovery. We have shown how a graph database can accommodate the results given by a variety of bioinformatics programs, and how Spfy is performant for data retrieval on the results for multiple analyses among over 10,000 genomes, providing results from big-data comparisons in the same efficiency as old analyses on single files.

I think this discussion sets the perfect tone for the big data applications provided by spfy. My only suggestion is to give examples of the types analysis that are mentioned in the review. One idea, instead of dropping "Impact on Public Heath Efforts" entirely, you can convert it into a use case example that highlights the benefit spfy provides.

5.1 Impact on Public Heath Efforts

The isolation and characterization of bacterial pathogens are critical for Public Health laboratories to rapidly respond to outbreaks, and to effectively monitor known and emerging pathogens through surveillance programs. Until recently, public-health agencies relied on laboratory tests such as serotyping, pulsed-field gel electrophoresis (PFGE), PCR-based amplification of known VFs, and disc-diffusion assays to identify AMR, for the characterization of bacterial isolates in outbreak, surveillance, and reference laboratory settings [1]. Current efforts are focused on predictive genomics, where the relevant phenotypic information can be determined through examination of the whole-genome sequence without need for the traditional laboratory test.

Spfy provides rapid and easy predictive genomic analyses of *E. coli* genomes while also addressing the problem of large scale comparisons. With the larger datasets involved in population genomics, it is no longer viable for individual researchers to download data to perform comparisons. Instead, efforts have focused on storing biological data online and enabling analyses on that data [34]. By using a graph database, Spfy can integrate results from different technologies including software predicted phenotypes as well as laboratory results and user-submitted metadata. In addition, datasets can be built and submitted from multiple labs for joint analysis.

5.2 Comparison with other bioinformatic pipeline technologies

The automated analyses of WGS is currently facilitated by existing scientific workflow technologies such as Galaxy [33]. Galaxy aims to provide a reproducible, computational interface which is accessible to individuals without programming knowledge. Galaxy defines a formal schema for linking different analysis software together, so the entire analysis pipeline can be replicated and also extended as new analysis tools are developed. The Galaxy worklow focus is on running an individual analysis pipeline. It does not include functionality to store and collate analysis results for large-scale comparative analysis.

The Bacterium Analysis Pipeline (BAP) [17] provides an integrated analysis pipeline for bacterial WGS data as a web service. It provides an individual per-genome report of the determined species, multilocus sequence type, VF and AMR genes [17].

Spfy is similar to these technologies in that it automates workflows for users, and as in Galaxy, uses task queues to distribute selected analyses. On a per file basis, Spfy performs at a similar speed to BAP on predictive genomics tasks, though Spfy does not provide genome assembly services. Spfy processes XXXX files over XX tasks in XXXX time, distributing computations over a task queue and multiple Docker compartmentalized containers. However, unlike these workflow managers, Spfy is designed to help solve the needless recomputation of analyses by storing results in a graph database for downstream comparative analysis. This allows Spfy to perform population-wide analyses, regardless of the individual analysis software used to generate the results for an individual genome.

Table 2: Comparison of various bioinformatic pipelines and their underlying database. Functionally, Spfy integrates different analysis modules as in BAP while also merging large datasets as in PATRIC.

	Spfy	Galaxy	BAP	PATRIC
Database	Blazegraph	PostgreSQL	MySQL + File System	MongoDB + Shock
Type	Graph	SQL	SQL + File System	NoSQL
Focus	Integrated Analyses	Workflow Technology	Batch Analysis	Integrated Analyses

PATRIC [18] adds support for comparing up to nine user-submitted genomes against a reference genome, targeting various gene annotations. PATRIC indexes a NoSQL document store to compare similar document types and is beneficial for comparative genomics. Unlike PARTIC, Spfy is targeted towards statistical comparisons of any population grouping, and Spfy's graph database has no limit on the number of genomes grouped for comparison. For example, Spfy can compare all genomes with a particular subtype for gene annotations, source metadata targets for gene annotations, or any two (or more) node or attribute types as shown in 5.

Maybe should include a comparison to PATRIC. Patric does offer some comparative genomic analysis, however they are not health focused (pathways, gene presence/absence etc), and you cannnot add your own data to the repository

6 CONCLUSIONS

Future work will focus on adding additional analyses modules to aid genotype to phenotype predictions using machine learning modules, and supporting bacterial species such as *Salmonella*, and *Campylobater*. While the integrated approach of storing and retrieving results provides enormous benefits, the developed analyses modules are self-contained and be transfered to existing platforms such as Galaxy. The source code for Spfy is hosted at `https://github.com/superphy/backend`, and is available for free under the open-source Apache 2.0 license. A developer guide is provided at `https://superphy.readthedocs.io/en/latest/`.

Conflict of interest. None declared.

7 ACKNOWLEDGEMENTS

References

[1] J Ronholm, Neda Nasheri, Nicholas Petronella, and Franco Pagotto. Navigating microbiological food safety in the era of whole-genome sequencing. *Clinical Microbiology Reviews*, 29(4):837–857, 2016.

[2] Birgitta Lytsy, Lars Engstrand, Åke Gustafsson, and Rene Kaden. Time to review the gold standard for genotyping vancomycin-resistant enterococci in epidemiology: Comparing whole-genome sequencing with pfge and mlst in three suspected outbreaks in sweden during 2013–2015. *Infection, Genetics and Evolution*, 2017.

[3] Kai Wang, Siu Tsan Yuen, Jiangchun Xu, Siu Po Lee, Helen HN Yan, Stephanie T Shi, Hoi Cheong Siu, Shibing Deng, Kent Man Chu, Simon Law, et al. Whole-genome sequencing and comprehensive molecular profiling identify new driver mutations in gastric cancer. *Nature genetics*, 46(6):573–582, 2014.

[4] Ryan KC Yuen, Bhooma Thiruvahindrapuram, Daniele Merico, Susan Walker, Kristiina Tammimies, Ny Hoang, Christina Chrysler, Thomas Nalpathamkalam, Giovanna Pellecchia, Yi Liu, et al. Whole-genome sequencing of quartet families with autism spectrum disorder. *Nature medicine*, 21(2):185–191, 2015.

[5] Laurel K Willig, Josh E Petrikin, Laurie D Smith, Carol J Saunders, Isabelle Thiffault, Neil A Miller, Sarah E Soden, Julie A Cakici, Suzanne M Herd, Greyson Twist, et al. Whole-genome sequencing for identification of mendelian disorders in critically ill infants: a retrospective analysis of diagnostic and clinical findings. *The Lancet Respiratory Medicine*, 3(5):377–387, 2015.

[6] Frederick E Dewey, Megan E Grove, Cuiping Pan, Benjamin A Goldstein, Jonathan A Bernstein, Hassan Chaib, Jason D Merker, Rachel L Goldfeder, Gregory M Enns, Sean P David, et al. Clinical interpretation and implications of whole-genome sequencing. *Jama*, 311(10):1035–1045, 2014.

[7] Andrew G McArthur, Nicholas Waglechner, Fazmin Nizam, Austin Yan, Marisa A Azad, Alison J Baylay, Kirandeep Bhullar, Marc J Canova, Gianfranco De Pascale, Linda Ejim, et al. The comprehensive antibiotic resistance database. *Antimicrobial agents and chemotherapy*, 57(7):3348–3357, 2013.

[8] Kortine Annina Kleinheinz, Katrine Grimstrup Joensen, and Mette Voldby Larsen. Applying the resfinder and virulencefinder web-services for easy identification of acquired antibiotic resistance and e. coli virulence genes in bacteriophage and prophage nucleotide sequences. *Bacteriophage*, 4(2):e27943, 2014.

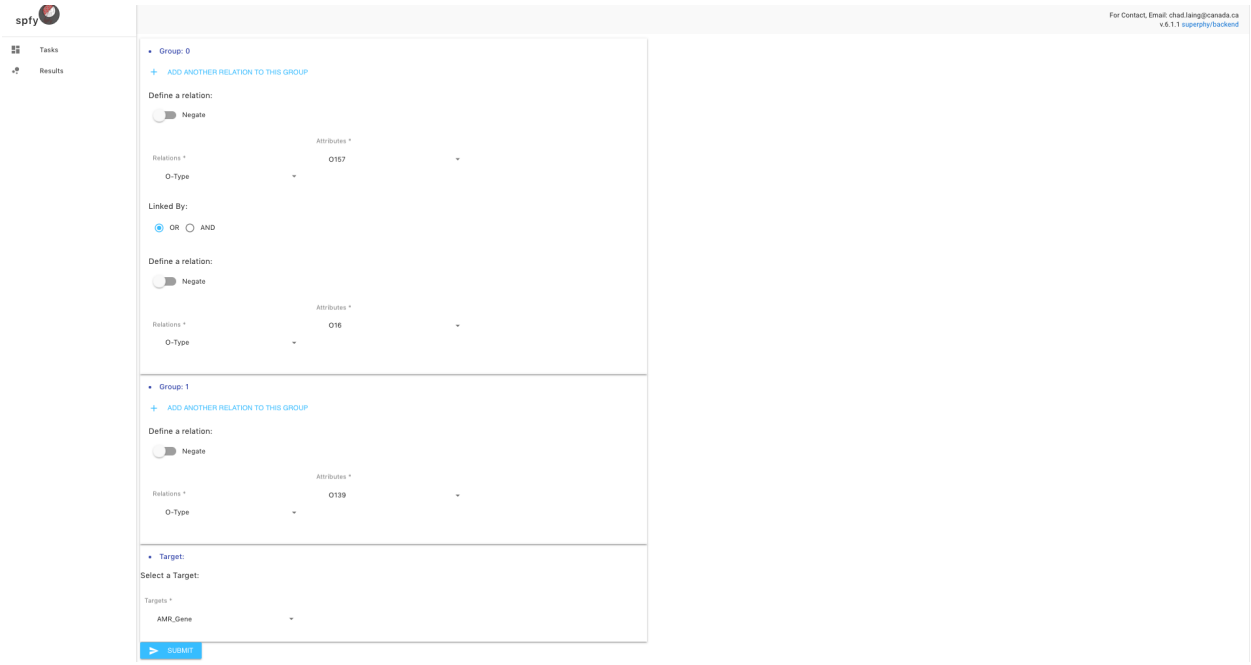


Figure 5: Genomes can be grouped by any node or attribute type for comparisons, and Spfy has no upper bound for the number of genomes in a group. Different types can also be joined into a group through logical connectives AND, OR, Negation, and any data type is also a valid target. This approach can be used to compare any data regardless of source software.

[9] Sushim Kumar Gupta, Babu Roshan Padmanabhan, Seydina M Diene, Rafael Lopez-Rojas, Marie Kempf, Luce Landraud, and Jean-Marc Rolain. Arg-annot, a new bioinformatic tool to discover antibiotic resistance genes in bacterial genomes. *Antimicrobial agents and chemotherapy*, 58(1):212–220, 2014.

[10] Martin Hunt, Alison E Mather, Leonor Sánchez-Busó, Andrew J Page, Julian Parkhill, Jacqueline A Keane, and Simon R Harris. Ariba: rapid antimicrobial resistance genotyping directly from sequencing reads. *Microbial genomics*, 3(10), 2017.

[11] Michael Inouye, Harriet Dashnow, Lesley-Ann Raven, Mark B Schultz, Bernard J Pope, Takehiro Tomita, Justin Zobel, and Kathryn E Holt. Srst2: Rapid genomic surveillance for public health and hospital microbiology labs. *Genome medicine*, 6(11):90, 2014.

[12] Dominic Lambert, Catherine D Carrillo, Adam G Koziol, Paul Manninger, and Burton W Blais. Genesippr: a rapid whole-genome approach for the identification and characterization of foodborne pathogens such as priority shiga toxigenic escherichia coli. *PLoS One*, 10(4):e0122928, 2015.

[13] Matthew D Whiteside, Chad R Laing, and Victor PJ Gannon. Phylotyper: in silico predictor of gene subtypes. *Bioinformatics*, 2017.

[14] Katrine G Joensen, Anna MM Tetzschner, Atsushi Iguchi, Frank M Aarestrup, and Flemming Scheutz. Rapid and easy in silico serotyping of escherichia coli using whole genome sequencing (wgs) data. *Journal of clinical microbiology*, pages JCM–00008, 2015.

[15] Danielle J Ingle, Mary Valcanis, Alex Kuzevski, Marija Tauschek, Michael Inouye, Tim Stinear, Myron M Levine, Roy M Robins-Browne, and Kathryn E Holt. In silico serotyping of e. coli from short read data identifies limited novel o-loci but extensive diversity of o: H serotype combinations within and between pathogenic lineages. *Microbial genomics*, 2(7), 2016.

[16] Catherine D Carrillo, Adam G Koziol, Amit Mathews, Noriko Goji, Dominic Lambert, George Huszczynski, Martine Gauthier, Kingsley Amoako, and Burton W Blais. Comparative evaluation of genomic and laboratory approaches for determination of shiga toxin subtypes in escherichia coli. *Journal of food protection*, 79(12):2078–2085, 2016.

[17] Martin Christen Frølund Thomsen, Johanne Ahrenfeldt, Jose Luis Bellod Cisneros, Vanessa Jurtz, Mette Voldby Larsen, Henrik Hasman, Frank Møller Aarestrup, and Ole Lund. A bacterial analysis platform: an integrated system for analysing bacterial whole genome sequencing data for clinical diagnostics and surveillance. *PloS one*, 11(6):e0157718, 2016.

[18] Alice R Wattam, James J Davis, Rida Assaf, Sébastien Boisvert, Thomas Brettin, Christopher Bun, Neal Conrad, Emily M Dietrich, Terry Disz, Joseph L Gabbard, et al. Improvements to patric, the all-bacterial bioinformatics database and analysis resource center. *Nucleic acids research*, 45(D1):D535–D542, 2016.

[19] Bala Swaminathan, Timothy J Barrett, Susan B Hunter, Robert V Tauxe, and CDC PulseNet Task Force. Pulsenet: the molecular subtyping network for foodborne bacterial disease surveillance, united states. *Emerging infectious diseases*, 7(3):382, 2001.

[20] Dennis A. Benson, Mark Cavanaugh, Karen Clark, Ilene Karsch-Mizrachi, David J. Lipman, James Ostell, and Eric W. Sayers. Genbank. *Nucleic Acids Research*, 41(D1):D36–D42, 2013.

[21] Matthew D Whiteside, Chad R Laing, Akiff Manji, Peter Kruczkiewicz, Eduardo N Taboada, and Victor PJ Gannon. Superphy: predictive genomics for the bacterial pathogen escherichia coli. *BMC microbiology*, 16(1):65, 2016.

[22] Chad Laing, Cody Buchanan, Eduardo N Taboada, Yongxiang Zhang, Andrew Kropinski, Andre Villegas, James E Thomas, and Victor PJ Gannon. Pan-genome sequence analysis using panseq: an online tool for the rapid analysis of core and accessory genomic regions. *BMC bioinformatics*, 11(1):461, 2010.

[23] Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, et al. Scikit-learn: Machine learning in python. *Journal of Machine Learning Research*, 12(Oct):2825–2830, 2011.

[24] Emma Griffiths, Damion Dooley, Morag Graham, Gary Van Domselaar, Fiona SL Brinkman, and William WL Hsiao. Context is everything: Harmonization of critical food microbiology descriptors and metadata for improved food safety and surveillance. *Frontiers in Microbiology*, 8:1068, 2017.

[25] Jerven T Bolleman, Christopher J Mungall, Francesco Strozzi, Joachim Baran, Michel Dumontier, Raoul JP Bonnal, Robert Buels, Robert Hoehndorf, Takatomo Fujisawa, Toshiaki Katayama, et al. Faldo: a semantic standard for describing the location of nucleotide and protein feature annotation. *Journal of biomedical semantics*, 7(1):39, 2016.

[26] Cátia Vaz, Alexandre P Francisco, Mickael Silva, Keith A Jolley, James E Bray, Hannes Pouseele, Joerg Rothganger, Mário Ramirez, and João A Carriço. Typon: the microbial typing ontology. *Journal of biomedical semantics*, 5(1):43, 2014.

[27] Tim Berners-Lee, James Hendler, Ora Lassila, et al. The semantic web. *Scientific american*, 284(5):28–37, 2001.

[28] Ian Horrocks, Bijan Parsia, Peter Patel-Schneider, and James Hendler. Semantic web architecture: Stack or two towers? In *International Workshop on Principles and Practice of Semantic Web Reasoning*, pages 37–41. Springer, 2005.

[29] Wes Felter, Alexandre Ferreira, Ram Rajamony, and Juan Rubio. An updated performance comparison of virtual machines and linux containers. In *Performance Analysis of Systems and Software (ISPASS), 2015 IEEE International Symposium On*, pages 171–172. IEEE, 2015.

[30] Torsten Seemann. Prokka: rapid prokaryotic genome annotation. *Bioinformatics*, 30(14):2068–2069, 2014.

[31] Samia N Naccache, Scot Federman, Narayanan Veeraraghavan, Matei Zaharia, Deanna Lee, Erik Samayoa, Jerome Bouquet, Alexander L Greninger, Ka-Cheung Luk, Barryett Enge, et al. A cloud-compatible bioinformatics pipeline for ultrarapid pathogen identification from next-generation sequencing of clinical samples. *Genome research*, 24(7):1180–1192, 2014.

[32] Alexandre G de Brevern, Jean-Philippe Meyniel, Cécile Fairhead, Cécile Neuvéglise, and Alain Malpertuy. Trends in it innovation to build a next generation bioinformatics solution to manage and analyse biological big data produced by ngs technologies. *BioMed research international*, 2015, 2015.

[33] Jeremy Goecks, Anton Nekrutenko, and James Taylor. Galaxy: a comprehensive approach for supporting accessible, reproducible, and transparent computational research in the life sciences. *Genome biology*, 11(8):R86, 2010.

[34] Michael C Schatz. Biological data sciences in genome research. *Genome research*, 25(10):1417–1422, 2015.

8 APPENDIX