16th Annual Bioinformatics Open Source Conference

BOSC 2015

Dublin, Ireland
July 10-11, 2015

http://www.open-bio.org/wiki/BOSC_2015

Welcome to BOSC 2015! The Bioinformatics Open Source Conference, established in 2000, is held every year as a Special Interest Group (SIG) meeting in conjunction with the Intelligent Systems for Molecular Biology (ISMB) Conference. BOSC is sponsored by the Open Bioinformatics Foundation (OBF), a non-profit group dedicated to promoting the practice and philosophy of Open Source software development and Open Science within the biological research community.

Sponsors

We are grateful to Google for their generous support for videorecording BOSC 2015, and we thank Curoverse (the team behind the open source platform Arvados) and GigaScience as returning sponsors. We also welcome Bina as a new sponsor.

BOSC 2015 Organizing Committee

Nomi Harris and **Peter Cock** (Co-Chairs)
Brad Chapman, Rob Davey, Chris Fields, Sarah Hird, Karsten Hokamp, Hilmar Lapp, Mónica Muñoz-Torres

Program Committee: Nomi Harris, Brad Chapman, Peter Cock, Karsten Hokamp, Raoul Bonnal, Chris Fields, Karen Cranston, Jens Lichtenberg, Eric Talevich, Frank Nothaft, Michael Heuer, Mónica Muñoz-Torres, Francesco Strozzi, Hans-Rudolf Hotz, Timothy Booth, Tiago Antão, George Githinji, Manuel Corpas, Thomas Down, Sarah Hird, Scott Markel, Rob Davey, Spencer Bliven, Michael Reich, Lorena Pantano, Björn Grüning, Hilmar Lapp, Daniel Blankenberg, Amye Kenall, Hervé Menager

BOSC is a community effort—we thank all those who made it possible, including the organizing committee, the program committee, the session chairs, our sponsors, and the ISMB SIG chair, Steven Leard.

If you are interested in helping to organize BOSC 2016, please email bosc@open-bio.org.

Talks and Posters

BOSC includes two full days of talks, posters, and Birds of a Feather interest groups (BOFs). Session topics this year include Data Science, Standards and Interoperability, Open Science and Reproducibility, Translational Bioinformatics, and Visualization, as well as the usual session on Bioinformatics Open Source Project Updates. We also have a special session this year for late-breaking lightning talks. The longer talks this year are 15 minutes (plus 5 minutes for questions); lightning talks are 5 minutes, with no time allocated for questions—we suggest that you find the lightning talk authors during the breaks to ask them questions.

This year's keynote speakers will be Holly Bik and Ewan Birney. Our panel topic this year is "**Open Source, Open Door: increasing diversity in the bioinformatics open source community**", with panelists Holly Bik, Michael R. Crusoe, Aleksandra Pawlik, Jason Williams and moderator Mónica Muñoz-Torres.

There are poster sessions both days that start during the lunch hour. Authors should put up their posters in their assigned poster spot before the first poster session (which starts at 13:00).

We have space for several last-minute posters, in addition to those listed in the program. Please contact us at bosc@open-bio.org if you'd like to present a last-minute poster.

Optional BOSC Dinner

We invite you to join BOSC organizers and attendees at a pay-your-own-way dinner the first evening of BOSC (Friday, 10 July, at 7pm) at Kennedy's Pub, 32 Westland Row, Dublin 2. Kennedy's is 1.3 km from the conference center—see http://url.ie/z1q1

If you want to join us for dinner, RSVP at http://bit.ly/BOSC2015-dinner before Friday at 3pm. The restaurant has space for 30 BOSC guests; only those who RSVP will be admitted.

In addition to the organized Friday dinner, there are always casual groups of BOSC attendees who go out to dinner the second day of the meeting.

OBF Membership

Professionals, scientists, students, and others active in open science or open source software or in the life sciences are invited to join BOSC's parent organization, the Open Bioinformatics Foundation (the OBF). The OBF grew out of the volunteer projects BioPerl, BioJava and Biopython and was formally incorporated in 2001 in order to handle modest requirements of hardware ownership, domain name management and funding for conferences and workshops. In 2005, we enacted bylaws for the first time, and along with it created a formal membership.

In 2012, upon overwhelming approval in a membership vote we changed from being independently incorporated to joining Software In The Public Interest, Inc., a fiscal sponsorship organization that aligns well with our own values and culture. We continue to maintain our own membership so that our community has a role in shaping our direction and vision. You can find information on how to join OBF on the OBF wiki at http://www.open-bio.org/wiki/Membership. There is no membership fee. Also, if you are interested in meeting and talking to some of the OBF Directors and members, please join us at the BOSC dinner (see above).

BOSC 2015 Schedule

Day 1 (Friday, 10 July 2015)

Time	Title	Speaker / Chair
7:30-9:00	**Registration**	
9:00-9:15	**Introduction and Welcome**	Nomi Harris and Peter Cock (Co-Chairs, BOSC 2015)
9:15-10:15	**Keynote: Bioinformatics: Still a scary world for biologists**	Holly Bik
10:15-10:45	***Coffee Break***	
10:45-12:30	**Session: Data Science**	Chair: Rob Davey
10:45-11:02	Apollo: Scalable & collaborative curation for improved comparative genomics	Mónica Muñoz-Torres
11:02-11:19	GOexpress: A R/Bioconductor package for the identification and visualisation of robust gene ontology signatures through supervised learning of gene expression data	Kévin Rue-Albrecht
11:19-11:36	Arvados: A Free Software Platform for Big Data Science	Peter Amstutz
11:36-11:53	Bringing Hadoop into Bioinformatics with Cloudgene and CloudMan	Sebastian Schoenherr
11:53-12:10	Segway: semi-automated genome annotation	Michael Hoffman
12:10-12:15	QualiMap 2.0: quality control of high throughput sequencing data	Konstantin Okonechnikov
12:15-12:20	A Genomics Virtual Laboratory	Andrew Lonie
12:20-12:25	BioSolr: Building better search for bioinformatics	Tony Burdett
12:25-12:30	Prioritization of structural variants based on known biological information	Brad Chapman
12:30-13:30	***Lunch***	
13:00-14:00	**Poster Session and Birds of a Feather** (overlapping with lunch)	
14:00-15:30	**Session: Standards and Interoperability**	Chair: Peter Cock
14:00-14:17	Portable workflow and tool descriptions with the CWL	Michael R. Crusoe
14:17-14:34	From peer-reviewed to peer-reproduced: a role for research objects in scholarly publishing in the life sciences	Alejandra Gonzalez-Beltran
14:34-14:51	Demystifying the Interoperability of Disparate Genomic Resources	Daniel Blankenberg
14:51-15:08	Increasing the utility of Galaxy workflows	John Chilton

Time	Title	Speaker / Chair
15:10-15:15	Kipper: A software package for sequence database versioning for Galaxy bioinformatics servers	Damion Dooley
15:15-15:20	Evolution of the Galaxy tool ecosystem - happier developers, happier users	Martin Čech
15:20-15:25	Bionode - Modular and universal bioinformatics	Bruno Vieira
15:25-15:30	The EDAM Ontology	Hervé Ménager
15:30-16:00	***Coffee Break***	
16:00-17:00	**Panel: Open Source, Open Door: increasing diversity in the bioinformatics open source community**	***Moderator***: Mónica Muñoz-Torres ***Panelists***: Holly Bik, Michael R. Crusoe, Aleksandra Pawlik, Jason Williams
17:00-17:10	Open Bioinformatics Foundation (OBF) Update	Hilmar Lapp (President, OBF)
17:10-17:15	**Announcements**	Nomi Harris
17:15-18:30	BOF/Unconference: Building successful open-source bioinformatics developer communities (Part 1)	Aidan Budd, Dave Clements, Manuel Corpas, Natasha Wood
17:15-18:30	**BOFs**	
19:00-	Pay-your-own-way BOSC dinner, Kennedy's	RSVP at bit.ly/BOSC2015-dinner (limited space--you must RSVP to attend)

Day 2 (Saturday, 11 July 2015)

Time	Title	Speaker or Session Chair
9:00-9:05	**Announcements**	Peter Cock and Nomi Harris
9:05-9:15	Codefest 2015 Report	Brad Chapman (Codefest 2015 Organizer)
9:15-10:15	**Keynote: Big Data in Biology**	Ewan Birney
10:15-10:45	***Coffee Break***	
10:45-12:30	**Session: Open Science and Reproducibility**	Chair: Mónica Muñoz-Torres
10:45-11:02	A curriculum for teaching Reproducible Computational Science bootcamps	Hilmar Lapp
11:02-11:19	Research shared: www.researchobject.org	Norman Morrison
11:19-11:36	Nextflow: a tool for deploying reproducible computational pipelines	Paolo Di Tommaso
11:36-11:53	Free beer today: how iPlant + Agave + Docker are changing our assumptions about reproducible science	John Fonner
11:55	The 500 builds of 300 applications in the HeLmod repository will at least get you started on a full suite of scientific applications	Aaron Kitzmiller
12:00	Bioboxes: Standardised bioinformatics tools using Docker containers.	Peter Belmann
12:05	The perfect fit for reproducible interactive research: Galaxy, Docker, IPython	Björn Grüning
12:10	COPO: Bridging the Gap from Data to Publication in Plant Science	Robert Davey
12:15	ELIXIR UK building on Data and Software Carpentry to address the challenges in computational training for life scientists	Aleksandra Pawlik
12:20	Parallel recipes: towards a common coordination language for scientific workflow management systems	Yves Vandriessche
12:25	openSNP - personal genomics and the public domain	Bastian Greshake
12:30-13:30	***Lunch***	
13:00-14:00	**Poster Session and BOFs (overlapping with lunch)**	
14:00-14:40	**Session: Translational Bioinformatics**	Chair: Brad Chapman
14:00-14:17	CIViC: Crowdsourcing the Clinical Interpretation of Variants in Cancer	Malachi Griffith
14:17-14:34	From Fastq To Drug Recommendation - Automated Cancer Report Generation using OncoRep & Omics Pipe	Tobias Meissner
14:35-14:40	Cancer Informatics Collaboration and Computation: Two	Ishwar

Time	Title	Speaker or Session Chair
	Initiatives of the U.S. National Cancer Institute	Chandramouliswaran
14:40-15:30	**Session: Bioinformatics Open Source Project Updates**	Chair: Nomi Harris
14:40-14:57	Biopython Project Update 2015	João Rodrigues
14:57-15:14	The biogems community: Challenges in distributed software development in bioinformatics	George Githinji and Pjotr Prins
15:14-15:31	Apache Taverna: Sustaining research software at the Apache Software Foundation	Stian Soiland-Reyes
15:30-16:00	***Coffee Break***	
16:00-16:30	**Session: Visualization**	Chair: Karsten Hokamp
16:00-16:17	Simple, Shareable, Online RNA Secondary Structure Diagrams	Peter Kerpedjiev
16:17-16:22	BioJS 2.0: an open source standard for biological visualization	Guy Yachdav
16:22-16:27	Visualising Open PHACTS linked data with widgets	Ian Dunlop
16:30-17:00	**Session: Late-Breaking Lightning Talks**	Chair: Hilmar Lapp
16:30	Biospectra-by-sequencing genetic analysis platform	Aurelie Laugraud
16:35	PhyloToAST: Bioinformatics tools for species-level analysis and visualization of complex microbial communities	Shareef Dabdoub
16:40	Otter/ZMap/SeqTools: A productive alternative to web browser genome visualisation	Gemma Guest
16:45	aRchive: enabling reproducibility of Bioconductor package versions	Nitesh Turaga
16:50	Developing an Arvados BWA-GATK pipeline	Pjotr Prins
16:55	Out of the box cloud solution for Next-Generation Sequencing analysis	Freerk van Dijk
17:00-17:10	Concluding Remarks	Nomi Harris and Peter Cock
17:15-18:15	BOF/Unconference: Building successful open-source bioinformatics developer communities (Part 2)	Aidan Budd, Dave Clements, Manuel Corpas, Natasha Wood
17:15-18:15	**BOFs**	

Any last-minute schedule updates will be posted at
http://www.open-bio.org/wiki/BOSC_2015_Schedule

Keynote Speakers

Holly Bik

Dr Holly Bik is a Birmingham Fellow (assistant professor) in the School of Biosciences at the University of Birmingham, UK. She obtained her Ph.D. in molecular phylogenetics at the University of Southampton, UK (working in conjunction with the Natural History Museum, London), followed by subsequent postdoctoral appointments at the Hubbard Center for Genome Studies at the University of New Hampshire and the UC Davis Genome Center.

Her research uses high-throughput environmental sequencing approaches (rRNA surveys, metagenomics) to explore biodiversity and biogeographic patterns in microbial eukaryote assemblages, with an emphasis on nematodes in marine sediments. Her long-term research aims to address existing bottlenecks encountered in –Omic analyses focused on microbial eukaryotes.

Holly's keynote talk topic is "Bioinformatics: Still a scary world for biologists".

> *Many biological disciplines remain staunchly traditional, where high-throughput DNA sequencing and bioinformatics have not yet become widely adopted. In this talk, I'll discuss the ongoing challenges and barriers facing biologists in the age of 'Omics, based on my experiences in transitioning from nematode taxonomy to computational biology research.*

Ewan Birney

Dr Ewan Birney is Joint Associate Director of EMBL-EBI, as well as Interim Head of the Centre for Therapeutic Target Validation. He played a vital role in annotating the genome sequences of the human, mouse, chicken and several other organisms. He led the analysis group for the ENCODE project, which is defining functional elements in the human genome. He was also one of the leaders of the BioPerl project.

He has received a number of prestigious awards including the 2003 Francis Crick Award from the Royal Society, the 2005 Overton Prize from the International Society for Computational Biology and the 2005 Benjamin Franklin Award for contributions in Open Source Bioinformatics. He was elected a Fellow of the Royal Society in 2014.

Ewan was a cofounder of the Open Bioinformatics Foundation, the organization that sponsors BOSC, and has been involved in BOSC since the first conference in 2000. He chaired the meeting in 2001, and gave one of the keynote talks in 2002. We are delighted to have him back as a keynote speaker for 2015.

Ewan's talk title is "Big Data in Biology".

> *Molecular biology is now a leading example of a data intensive science, with both pragmatic and theoretical challenges being raised by data volumes and dimensionality of the data. These changes are present in both "large scale" consortia science and small scale science, and across now a broad range of applications – from human health, through to agriculture and ecosystems. All of molecular life science is feeling this effect. This shift in modality is creating a wealth of new opportunities and has some accompanying challenges. In particular there is a continued need for a robust information infrastructure for molecular biology. This ranges from the physical aspects of dealing with data volume through to the more statistically challenging aspects of interpreting it. A particular problem is finding causal relationships in the high level of correlative data. Genetic data are particular useful in resolving these issues. I will end with the serendipitous invention of using DNA for an entirely different reason – as a long-time horizon digital archiving material. I will describe this method and some of its benefits (as well as a few downsides) and explain how a future culture in 10,000 years time may still be able to read all of Shakespeare's sonnets – and perhaps much more.*

Panel: *Open Source, Open Door: increasing diversity in the bioinformatics open source community*

Every year, BOSC includes a panel discussion that offers all attendees the chance to engage in conversation with the panelists and each other. This year's panel focuses on the important topic of what can be done to increase the diversity of participants in BOSC and in open source bioinformatics in general. The panel chair and panelists are:

Panel chair **Mónica Muñoz-Torres** (@monimunozto) is the lead biocurator at Berkeley Bioinformatics Open-Source Projects (BBOP). She co-leads the Community Curation group within the global initiative to sequence and annotate the genomes of 5,000 arthropods (i5K Initiative), and is a member of the Executive Committee of the International Society for Biocuration (ISB). As a graduate student, Monica founded the first Southeastern Chapter of the Society for Advancement of Hispanics/Chicanos and Native Americans in Science (SACNAS) at Clemson University. She is currently working on forming the first professional chapter of SACNAS in the San Francisco Bay area.

Holly Bik (@hollybik) is a Birmingham Fellow (assistant professor) in the School of Biosciences at the University of Birmingham, UK. Her research uses high-throughput environmental sequencing approaches (rRNA surveys, metagenomics) to explore biodiversity and biogeographic patterns in microbial eukaryote assemblages. Holly's efforts to promote diversity include serving as an invited speaker at the Girls Who Code initiative and leading the organization of bioinformatics workshops for undergraduate students at minority-serving institutions (including Historically Black Colleges).

Michael R. Crusoe (@biocrusoe) is the lead for the k-h-mer project at C. Titus Brown's Lab for Data Intensive Biology at the University of California, Davis in the School of Veterinary Medicine. A community-minded bioinformatics research software engineer and Software Carpentry instructor, he is also a member of the Debian Med software packaging team. Michael's social justice background includes a prior seat on the board for the Phoenix, Arizona chapter of GLSEN, the Gay, Lesbian, and Straight Education Network and he is proud to be a supporter of the Ada Initiative.

Aleksandra Pawlik (@aleksandrana) is a Training Lead at the Software Sustainability Institute at Manchester University, UK. She is a member of the Steering Committees for Data Carpentry and Software Carpentry Foundation. Currently, Aleksandra is collaborating on training with the ELIXIR project supporting the bioinformatics community. As a certified Software and Data Carpentry instructor Aleksandra has taught at a number of workshops, including Software Carpentry for Women in Science and Engineering, which she co-organised.

Jason Williams (@JasonWilliamsNY) is the Lead of the iPlant Collaborative's Education, Outreach, Training (EOT) group, based at Cold Spring Harbor Laboratory, where he has worked for over 10 years. He is also a Lead Instructor of "The Science Institute" at Yeshiva University High School for Girls, and the Treasurer of the Software Carpentry Foundation. Diversity is a focus of Jason's work at the DNA Learning Center and with iPlant, where he works to target outreach along the entire spectrum of underrepresented and underserved groups ranging from minorities in urban communities to first-generation college students at rural institutions.

In addition to the panelists listed above, the BOSC 2015 co-chairs **Nomi Harris** and **Peter Cock** will be on hand, along with other Open Bioinformatics Foundation (OBF) Board Members and BOSC organising committee members, to comment on what BOSC and the OBF are doing to try to improve diversity in the open source bioinformatics community, and to listen to suggestions and feedback.

Talk and Poster Abstracts

In the pages that follow, talk abstracts appear in the order in which the talks will be presented. Some authors will also present their work as posters. Those abstracts have a poster number above the abstract. Poster-only abstracts appear after the talk abstracts.

There are also a few spaces available for last-minute posters. If you would like to present one, please email your abstract (which must meet the BOSC criteria of freely available source and recognized open source license) to bosc@open-bio.org.

Title	Author	Poster #
Apollo: Scalable & collaborative curation for improved comparative genomics	Mónica Muñoz-Torres	P1
GOexpress: A R/Bioconductor package for the identification and visualisation of robust gene ontology signatures through supervised learning of gene expression data	Kévin Rue-Albrecht	P2
Arvados: A Free Software Platform for Big Data Science	Peter Amstutz	-
Bringing Hadoop into Bioinformatics with Cloudgene and CloudMan	Sebastian Schoenherr	-
Segway: semi-automated genome annotation	Michael Hoffman	P3
QualiMap 2.0: quality control of high throughput sequencing data	Konstantin Okonechnikov	P4
A Genomics Virtual Laboratory	Andrew Lonie	P5
BioSolr: Building better search for bioinformatics	Tony Burdett	P6
Prioritization of structural variants based on known biological information	Brad Chapman	-
Portable workflow and tool descriptions with the CWL	Michael R. Crusoe	P7
From peer-reviewed to peer-reproduced: a role for research objects in scholarly publishing in the life sciences	Alejandra Gonzalez-Beltran	-
Demystifying the Interoperability of Disparate Genomic Resources	Daniel Blankenberg	-
Increasing the utility of Galaxy workflows	John Chilton	-
Kipper: A software package for sequence database versioning for Galaxy bioinformatics servers	Damion Dooley	-
Evolution of the Galaxy tool ecosystem - happier developers, happier users	Martin Čech	P8
Bionode - Modular and universal bioinformatics	Bruno Vieira	P9
The EDAM Ontology	Hervé Ménager	P10
A curriculum for teaching Reproducible Computational Science bootcamps	Hilmar Lapp	-
Research shared: www.researchobject.org	Norman Morrison	P11
Nextflow: a tool for deploying reproducible computational pipelines	Paolo Di Tommaso	P12
Free beer today: how iPlant + Agave + Docker are changing our assumptions about reproducible science	John Fonner	P13
The 500 builds of 300 applications in the HeLmod repository will at least get you started on a full suite of scientific applications	Aaron Kitzmiller	-
Bioboxes: Standardised bioinformatics tools using Docker containers.	Peter Belmann	P14
The perfect fit for reproducible interactive research: Galaxy, Docker, IPython	Björn Grüning	-
COPO: Bridging the Gap from Data to Publication in Plant Science	Robert Davey	P15
ELIXIR UK building on Data and Software Carpentry to address the challenges in computational training for life scientists	Aleksandra Pawlik	P16

Parallel recipes: towards a common coordination language for scientific workflow management systems	Yves Vandriessche	P17
openSNP - personal genomics and the public domain	Bastian Greshake	P18
CIViC: Crowdsourcing the Clinical Interpretation of Variants in Cancer	Malachi Griffith	P19
From Fastq To Drug Recommendation - Automated Cancer Report Generation using OncoRep & Omics Pipe	Tobias Meissner	P20
Cancer Informatics Collaboration and Computation: Two Initiatives of the U.S. National Cancer Institute	Ishwar Chandra-mouliswaran	P21
Biopython Project Update 2015	João Rodrigues	-
The biogems community: Challenges in distributed software development in bioinformatics	George Githinji and Pjotr Prins	-
Apache Taverna: Sustaining research software at the Apache Software Foundation	Stian Soiland-Reyes	P22
Simple, Shareable, Online RNA Secondary Structure Diagrams	Peter Kerpedjiev	-
BioJS 2.0: an open source standard for biological visualization	Guy Yachdav	P23
Visualising Open PHACTS linked data with widgets	Ian Dunlop	P24
Biospectra-by-sequencing genetic analysis platform	Aurelie Laugraud	P25
PhyloToAST: Bioinformatics tools for species-level analysis and visualization of complex microbial communities	Shareef Dabdoub	-
Otter/ZMap/SeqTools: A productive alternative to web browser genome visualisation	Gemma Guest	P26
aRchive: enabling reproducibility of Bioconductor package versions	Nitesh Turaga	P27
Developing an Arvados BWA-GATK pipeline	Pjotr Prins	P28
Out of the box cloud solution for Next-Generation Sequencing analysis	Freerk van Dijk	P29
Poster only:		
Aequatus: Visualising complex similarity relationships among species	Anil Thanki	P30
MOLGENIS Workbench for Systems Medicine	K. Joeri van der Velde	P31
SPINGO: a rapid species-classifier for microbial amplicon sequences	Feargal Ryan	P32
ANNOgesic - A computational pipeline for RNA-Seq based transcriptome annotations of bacteria	Konrad Förstner	P33
BioXSD — a data model for sequences, alignments, features, measured and inferred values	Matúš Kalaš	P34
MGkit: A Metagenomic Framework For The Study Of Microbial Communities	Francesco Rubino	P35
From scaffold to submission in a day: a new software pipeline for rapid genome annotation and analysis	Sascha Steinbiss	P36
Walk-in Posters		P36-44

This talk is accompanied by poster #1.

Apollo: Scalable & collaborative curation for improved comparative genomics

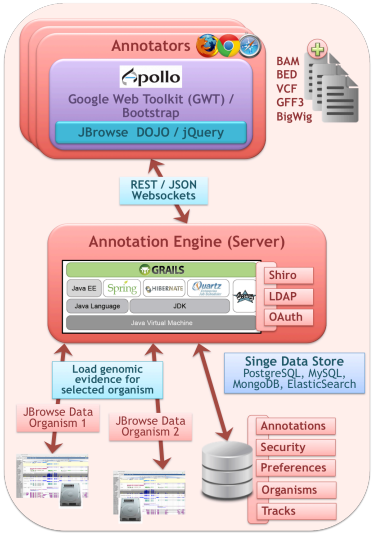
Monica C Munoz-Torres[1], Nathan A Dunn[1], Deepak Unni[2], Seth Carbon[1], Colin Diesh[2], Heiko Dietze[1], Christopher Mungall[1], Nicole Washington[1], Christine E Elsik[2], Ian Holmes[3], and Suzanna E Lewis[1]

[1] Lawrence Berkeley National Laboratory, Genomics Division, Berkeley, CA. Email: MCMunozT@lbl.gov
[2] University of Missouri, Divisions of Plant and Animal Sciences, Columbia, MO.
[3] University of California Berkeley, Bioengineering, Berkeley, CA.

Project Website: http://genomearchitect.org/
Source Code: https://github.com/GMOD/Apollo
License: Berkeley Software Distribution (BSD) License. See https://github.com/GMOD/Apollo/blob/master/LICENSE.md

Comparative genome analysis requires high quality annotations of all genomic elements. Today's sequencing projects face numerous challenges including lower coverage, more frequent assembly errors, and the lack of closely related species with well-annotated genomes. Apollo is a web-based application that supports and enables collaborative genome curation in real time, analogous to Google Docs, allowing teams of curators to improve on existing automated gene models through an intuitive interface.

Apollo's architecture is built on top of the JBrowse framework and is composed of a web-based client, an annotation-editing engine, and a server-side data service. It allows users to visualize automated gene models, protein alignments, expression and variant data, and with these, conduct structural and/or functional annotations. To support the diverse needs of a growing community, we have recently completed two major efforts to improve functionality and performance: 1) Significant architectural changes to adopt the Grails JVM framework, and 2) Adoption of a queryable datastore to house annotations. The improved architecture allows users to more easily query the data and build extensions, supports multiple organisms per server, and also allows a larger set of sequence annotations based on the Sequence Ontology. A more flexible user interface via a removable side-dock provides improved search functionality, validation checks, and editing capability, and offers fine-grained user and group level permission.



Researchers from nearly one hundred institutions worldwide are currently using Apollo for distributed curation efforts in over sixty genome projects across the tree of life: from plants to arthropods, to fungi, to species of fish and other vertebrates including human, cattle (bovine), and dog. We are training the next generation of researchers by reaching out to educators to make these tools available as part of curricula, offering workshops and webinars to the scientific community, and through widely applied systems such as iPlant and DNA Subway. We are currently integrating Apollo into an annotation environment combining gene structural and functional annotation, transcriptomic, proteomic, and phenotypic annotation. In this presentation we will describe in detail its utility to users, introduce the new architecture, and offer details of our future plans.

BOSC 2015 – *Data Science*
Day One, 10 July 2015, 10:45 – 12:30
Session chair: Rob Davey

This talk is accompanied by poster #2.

GOexpress: A R/Bioconductor package for the identification and visualisation of robust gene ontology signatures through supervised learning of gene expression data

Kévin Rue-Albrecht[1], Paul A. McGettigan[1], Belinda Hernández[2,4], Nicolas, C. Nalpas[1], David A. Magee[1], Andrew C. Parnell[2], Stephen V. Gordon[3,4] and David E. MacHugh[1,4]

[1] Animal Genomics Laboratory, UCD School of Agriculture and Food Science, University College Dublin, Dublin 4, Ireland. Email: kevin.rue@ucdconnect.ie

[2] UCD School Of Mathematical Sciences, University College Dublin, Dublin 4, Ireland.

[3] UCD School of Veterinary Medicine, University College Dublin, Dublin 4, Ireland.

[4] UCD Conway Institute of Biomolecular and Biomedical Research, University College Dublin, Dublin 4, Ireland.

Project Website: http://www.bioconductor.org/packages/release/bioc/html/GOexpress.html
Source Code: https://github.com/kevinrue/GOexpress
License: GPL (>= 3)

Main Text of Abstract

Background

The standardisation and decreasing cost of transcriptomics platforms has allowed for more complex experimental setups, including multiple experimental factors and levels. Identification of gene expression profiles that differentiate experimental groups is critical for discovery and analysis of key molecular pathways and also selection of robust diagnostic or prognostic biomarkers. While integration of differential expression statistics has been proposed to inform gene set enrichment analyses, such approaches are typically limited to single gene lists resulting from two-group comparisons or time-series analyses.

Results

We introduce GOexpress, a software package for scoring and summarising the ability of ontology-related genes to simultaneously classify samples from multiple experimental groups. GOexpress integrates normalised gene expression data (*e.g.* from microarray and RNA-seq experiments) and phenotypic information of individual samples with gene ontology annotations to derive a ranking of genes and gene ontologies using a supervised learning approach. The default random forest algorithm allows interactions between all experimental factors, and competitive scoring of expressed gene features to evaluate their relative importance in clustering the predefined groups of samples.

Conclusions

GOexpress enables rapid identification and visualisation of robust ontology-related gene panels that robustly classify groups of samples, and supports both categorical (e.g. infection, treatment) and continuous (e.g. time-series, drug concentrations) experimental factors. The use of standard Bioconductor extension packages and publicly available gene ontology annotations facilitates straightforward integration of GOexpress within existing analytical pipelines.

BOSC 2015 – *Data Science*
Day One, 10 July 2015, 10:45 – 12:30
Session chair: Rob Davey

Arvados: A Free Software Platform for Big Data Science

Peter Amstutz <peter.amstutz@curoverse.com>, Brett Smith <brett@curoverse.com>,
Ward Vandewege <ward@curoverse.com>, Tom Clegg <tom@curoverse.com>,
Radhika Chippada <radhika@curoverse.com>, Alexander Zaranek <awz@curoverse.com>
Curoverse, Inc.

http://arvados.org http://github.com/curoverse/arvados Affero GPL v3, Apache v2

Large-scale bioinformatics such as genomics requires the application of cluster computing, with many nodes working in parallel to produce results in a reasonable amount of time. When a compute job draws on terabytes of data, uses days compute time, and produces thousands of files, robust management of data sets and the analysis tools used on them is essential to avoid errors that may lead lead to wasted effort or invalid results. To best serve the needs of science, computing platforms should be designed from the ground up to achieve data integrity, provenance, and computational reproducibility.

This talk will introduce the Arvados (http://arvados.org) platform for data science. Arvados is a software system for managing compute clusters built around a scale-out content-addressed distributed file system (Arvados Keep) for storage, a cluster job queuing system designed for reproducibility (Arvados Crunch), and a user and group permission system for controlling and sharing access to those resources. Arvados provides web based and command line tools for transferring, managing, sharing, and computing on very large data sets.

Arvados is designed to scale from a single laptop to cluster and cloud based deployments with dozens of nodes. Arvados is also designed to federate with other Arvados instances, with easy transfer of data and computation between instances. For example, only a single command “arv-copy” is required to copy a complex computation pipeline from a laptop to a cluster or cloud instance (or between instances), where that computation can be run immediately with no additional provisioning or configuration on the target system. The Arvados project is also a founding member of the Common Workflow Language working group, and provides robust support for running computational workflows that are portable across multiple vendor platforms.

This talk will describe the Arvados architecture, describe how Arvados has been used successfully in research, and how interested participants can download and try Arvados for themselves and join the community. Arvados is free software, with services licensed under the GNU Affero General Public License version 3, with SDKs under the Apache License 2.0.

BOSC 2015 – *Data Science*
Day One, 10 July 2015, 10:45 – 12:30
Session chair: Rob Davey

Bringing Hadoop into Bioinformatics with Cloudgene and CloudMan

Sebastian Schönherr[1], Lukas Forer[1], Davor Davidović[3], Hansi Weissensteiner[1], Florian Kronenberg[1], Enis Afgan[2,3]

[1] Division of Genetic Epidemiology; Department of Medical Genetics, Molecular and Clinical Pharmacology; Innsbruck Medical University, Innsbruck, Austria. *Email*: sebastian.schoenherr@i-med.ac.at

[2] Department of Biology, Johns Hopkins University, Baltimore, MD, USA

[3] Centre for Informatics and Computing, Ruđer Bošković Institute (RBI), Zagreb, Croatia

Project Website: http://cloudman.irb.hr; http://cloudgene.uibk.ac.at
Source Code: https://github.com/galaxyproject/cloudman; https://github.com/genepi/cloudgene
License: MIT (CloudMan); GPL (Cloudgene)
Acknowledgment: EU FP7 project (grant agreement 602133); Michigan Imputation Server Team

Despite the evident potential of the MapReduce model and existence of bioinformatic algorithms and applications, those are still to become widely adopted in the bioinformatics data analysis. The Hadoop MapReduce model offers a simple framework for data parallelism by providing automated runtime recovery (for both task runtime and hardware failures), implicit scalability (tasks automatically run in parallel batch mode), as well as data replication and locality (reduce data movement, hence increase processing capacity). We identify two prerequisites for wider adoption and higher utilization of MapReduce tools: (1) abstract the technical details of how multiple existing MapReduce tools are composed, and (2) provide easy access to the necessary compute infrastructure and the appropriate environment. Satisfying these requirements would allow bioinformatics domain experts to focus on the analysis while the required technical details are hidden.

At BOSC 2012, two platforms were presented: Cloudgene - a MapReduce tool execution platform leveraging Hadoop, and CloudMan - a cloud resource manager. Since then, we have combined and extended these two platforms to provide a readily available and an accessible Hadoop-based bioinformatics environment for the Cloud. Cloudgene, other than allowing arbitrary MapReduce tools to be integrated and used to craft an analysis, has been extended as a job execution engine for currently two dedicated services: an imputation service developed in cooperation with the Center for Statistical Genetics, University of Michigan (available at *imputationserver.sph.umich.edu*) and a mtDNA analysis service (available at *mtdna-server.uibk.ac.at*). Thus far, the “Michigan Imputation Server” has shown remarkable popularity and scalability with over 690,000 human genomes being imputed within one year. These services have been deployed on dedicated hardware and offer a simple interface for the specific tasks while the jobs are being executed in the MapReduce fashion. This demonstrates a positive disposition towards wider adoption of MapReduce paradigm in the bioinformatics data analysis space given accessible and effective solutions.

To facilitate easy access to such MapReduce solutions for bioinformatics and broaden the availability of these services, we have extended CloudMan to provide a Hadoop-based environment with pre-configured Cloudgene. CloudMan handles the tasks of procuring required cloud resources and configuring the appropriate environment, thus insulating the user from the low-level technical details otherwise required. Because CloudMan is compatible with multiple cloud technologies, it is now feasible to deploy this environment on a range of private and public clouds. This makes it possible for anyone to obtain a scalable Hadoop-based cluster with Cloudgene pre-installed and readily execute MapReduce tools.

This talk will present the motivation for supporting greater adoption of MapReduce-based applications in the bioinformatics data analysis space followed by the details of the described services and their functionality.

BOSC 2015 – *Data Science*
Day One, 10 July 2015, 10:45 – 12:30
Session chair: Rob Davey

This talk is accompanied by poster #3.

Segway: semi-automated genome annotation

Eric Roberts[1], Michael M. Hoffman[1,2,3]

[1] Princess Margaret Cancer Centre, Toronto, ON, Canada.

[2] Department of Medical Biophysics, University of Toronto, Toronto, ON, Canada.

[3] Department of Computer Science, University of Toronto, Toronto, ON, Canada. Email: michael.hoffman@utoronto.ca

Project Website: http://segway.hoffmanlab.org/
Source Code: https://bitbucket.com/hoffmanlab/segway/
License: GNU General Public License version 2

The free Segway software package contains a method for analyzing multiple tracks of functional genomics data. Our method uses a dynamic Bayesian network (DBN) model, which enables it to analyze the entire genome at 1-bp resolution even in the face of heterogeneous patterns of missing data. This method is the first application of DBN techniques to genome-scale data and the first genomic segmentation method designed for use with the maximum resolution data available from ChIP-seq experiments without downsampling. Our software has extensive documentation and was designed from the outset with external users in mind. Segway annotations for the human epigenome are now built-in to the Ensembl and UCSC Genome Browsers.

We have continued development of Segway and Segway annotations since our initial publication (Hoffman et al. 2012 *Nat Methods* 9:473). We have switched to open development in a Mercurial repository on Bitbucket. Improvements in deployment of Segway and the underlying Graphical Models Toolkit (GMTK) and Genomedata functional data storage system mean user installation is now substantially easier. We use the drone.io continuous integration system and a series of new regression tests to automatically ensure a high level of software quality.

We have added a number of new features to Segway. It now includes support for running in a local mode without a cluster, in addition to the Grid Engine, Platform Load Sharing Facility (LSF), and Portable Batch System (PBS) cluster management systems. Segway now supports a concatenated segmentation mode for analyzing multiple grouped datasets. We continue to develop other new features such as a hierarchical segmentation mode.

BOSC 2015 – *Data Science*
Day One, 10 July 2015, 10:45 – 12:30
Session chair: Rob Davey

This talk is accompanied by poster #4.

QualiMap 2.0: quality control of high throughput sequencing data

Konstantin Okonechnikov[1], Ana Conesa[2], Fernando García-Alcalde[1,3]

[1] Department of Molecular Biology, Max Planck Institute for Infection Biology, D-10117, Berlin, Germany. Email: okonechnikov@mpiib-berlin.mpg.de

[2] Bioinformatics and Genomics Department, Centro de Investigación Príncipe Felipe, 46012, Valencia, Spain

[3] Roche Pharma Research and Early Development, Roche Innovation Center Basel, Switzerland

Project Website: http://qualimap.org/
Source Code: http://bitbucket.org/kokonech/qualimap/
License: GNU General Public License v2

Main Text of Abstract

Detection of random errors and systematic biases is a crucial step of a robust pipeline for processing of high throughput sequencing (HTS) data. There are bioinformatics software tools capable of performing this task. Some of them are suitable for general analysis of HTS data while others are targeted to a specific sequencing technology. Qualimap 2.0 represents a next step in the QC analysis of HTS data. It is a multiplatform user-friendly application with both graphical user and command line interfaces.

Qualimap includes four analysis modes: **BAM QC**, **Counts QC**, **RNA-seq QC** and **Multi-sample BAM QC**. Based on the selected type of analysis, users provide input data in the form of a BAM/SAM alignment, GTF/GFF/BED annotation file and/or read counts table. The results of the QC analysis are presented as an interactive report from GUI, as a static report in HTML or PDF format and as a plain text file suitable for parsing and further processing. The latter two analysis modes are first introduced in version 2.0. **Multi-sample BAM QC** allows combined quality control estimation for multiple alignment files. For this purpose Qualimap combines **BAM QC** results from multiple samples and creates a number of plots summarizing the datasets. **RNA-seq QC** performs computation of metrics specific for RNA-seq data, including per-transcript coverage, junction sequence distribution and reads genomic localization.

In addition, a large number of fixes and enhancements were implemented since the first version of Qualimap was released. Most of the bugs were reported by the users. Additionally due to open-source access to the code several fixes were implemented by the users and accepted in the main repository.

BOSC 2015 – *Data Science*
Day One, 10 July 2015, 10:45 – 12:30
Session chair: Rob Davey

This talk is accompanied by poster #5.

A Genomics Virtual Laboratory

Enis Afgan[1], Clare Sloggett[2], Nuwan Goonasekera[2], Michael Pheasant[3], Ron Horst[3], Mark Crowe[4], Igor Manukin[3], Simon Gladman[2], Yousef Kowsar[2], Derek Benson[3], Andrew Lonie[2,5]

[1]Johns Hopkins University, USA; [2]University of Melbourne, Australia; [3]University of Queensland, Australia; [4]Queensland Facility for Advanced Bioinformatics, Australia; [5]alonie@unimelb.edu.au

Project URL: http://genome.edu.au
Code URL: https://github.com/galaxyproject/galaxy-cloudman-playbook

Background
Analyzing high throughput genomics data is a complex and compute intensive task, generally requiring numerous software tools and large reference data sets, tied together in successive stages of data transformation and visualization. A computational platform enabling best practice genomics analysis ideally meets a number of requirements, including: a wide range of analysis and visualisation tools, closely linked to large user and reference data sets; workflow platform(s) enabling accessible, reproducible, portable analyses, through a flexible set of interfaces; highly available, scalable computational resources; and flexibility and versatility in the use of these resources to meet demands and expertise of a variety of users. Access to an appropriate computational platform can be a significant barrier to researchers, as establishing such a platform requires a large upfront investment in hardware, experience, and expertise.

Results
We designed and implemented the Genomics Virtual Laboratory (GVL) as a middleware layer of machine images, cloud management tools, and online services that enable researchers to build arbitrary sized compute clusters on demand, pre-populated with fully configured bioinformatics tools, reference datasets and workflow and visualisation options. The platform is flexible in that users can conduct analyses through web-based (Galaxy, RStudio, IPython Notebook) or command-line interfaces, and add/remove compute nodes and data resources as required. Best practice tutorials and protocols provide a path from introductory training to practice. The GVL is available on the OpenStack-based Australian Research Cloud (http://nectar.org.au) and the Amazon Web Services cloud. The principles, implementation and build process are designed to be cloud agnostic.

Conclusion
We provide a blueprint for the design and implementation of a cloud-based Genomics Virtual Laboratory. We discuss scope, design considerations and technical and logistical constraints, and explore the value added to the research community through the suite of services and resources provided by our implementation.

BOSC 2015 – *Data Science*
Day One, 10 July 2015, 10:45 – 12:30
Session chair: Rob Davey

This talk is accompanied by poster #6.

BioSolr: Building better search for bioinformatics

Tony Burdett[1], Matt Pearce[2], Tom Winch[2], Charlie Hull[2], Helen Parkinson[3] and Sameer Velankar[3]

[1] European Bioinformatics Institute (EMBL-EBI), European Molecular Biology Laboratory, Wellcome Trust Genome Campus, Hinxton, Cambridge CB10 1SD, United Kingdom: tburdett@ebi.ac.uk

[2] Flax, St Johns Innovation Centre, Cowley Road, Cambridge, CB4 0WS, United Kingdom

[3] European Bioinformatics Institute (EMBL-EBI), European Molecular Biology Laboratory, Wellcome Trust Genome Campus, Hinxton, Cambridge CB10 1SD, United Kingdom.

BioSolr Wiki: http://tinyurl.com/biosolr
Source Code: https://github.com/flaxsearch/BioSolr
License: http://www.apache.org/licenses/LICENSE-2.0
Mailing list: solr-users@ebi.ac.uk

Data retrieval is common in bioinformatics databases, however, optimal strategies for indexing rich biomedical data are less well understood. Biomedical data often contains hierarchical components, such as annotations to ontologies, and therefore do not conform to the flattened document-based model imposed by most search technologies. BioSolr advances the state of the art with regard to indexing and querying biomedical data with open source software. This unique BBSRC funded collaboration between Flax, an open source search specialist company based in Cambridge, and The European Bioinformatics Institute (EMBL-EBI), brings together experts in biological data management and experts in utilising the world-leading Apache Lucene/Solr search engine framework to address the challenges of making biomedical data more accessible. Challenges include integrating ontology-enabled search and searching by common classification systems (taxonomy, enzyme classifications, protein families etc).

BioSolr is developing software to facilitate indexing of ontologies, ontology driven faceting, searching Solr indexes with SPARQL, FASTA Solr search components, and an "x-join" search component to integrate external data resources with Solr. Development is in the form of Solr patches, plugins or clients, and all code developed as part of the BioSolr project is available as open source software on GitHub. In addition, BioSolr is building a wide community of users of Lucene/Solr for bioinformatics via international workshops and the open source search developer community. BioSolr is also working to identify a series of best practices for working with Solr in bioinformatics. Requirements and common usage scenarios across the full spectrum of bioinformatics have been collated and cover a range of domains, including searching over protein structures and sequences, ontologies and literature.

BOSC 2015 – *Data Science*
Day One, 10 July 2015, 10:45 – 12:30
Session chair: Rob Davey

Title	Prioritization of structural variants based on known biological information
Authors	*Brad Chapman*, Rory Kirchner, Miika Ahdesmaki, Justin Johnson, Shannan Ho Sui, Oliver Hofmann
Affiliations	Harvard Chan School Bioinformatics Core (http://hsphbio.ghost.io/), AstraZeneca Oncology (http://www.astrazeneca.com/Medicines/Oncology), Wolfson Wohl Cancer Research Centre (http://www.gla.ac.uk/researchinstitutes/cancersciences/ics/facilities/wwcrc/)
Contact	bchapman@hsph.harvard.edu
Availability	https://github.com/chapmanb/bcbio-nextgen
License	MIT

High-throughput human resequencing characterizes whole genome changes with the goal of linking variations to disease, drug responses or other phenotypes. The primary challenge following sensitive and precise variant detection is prioritizing the large number of results in the context of previously known biological information. This is especially problematic for samples that are not well explained by short variations like single nucleotide polymorphisms (SNPs) or small insertions and deletions. In these cases, structural variations such as larger insertions, deletions, rearrangments or copy number variations (CNVs) provide additional sources of causative variability. However, detecting structural variations from short reads is challenging, so biologists must search through a noisier dataset to find potentially relevant mutations for additional investigation.

We'll discuss an approach to help prioritize structural variations using pre-existing biological information. The approach is general and only reliant on inputs that link known mutations to genomic position, allowing incorporation of custom BED or VCF files into analyses. In this talk, we'll emphasize using public databases like COSMIC (https://cancer.sanger.ac.uk/cosmic), ClinVar (http://www.ncbi.nlm.nih.gov/clinvar/) and CIViC (https://civic.genome.wustl.edu/) to evaluate cancer samples. We overlay known variants with existing annotations on genes, domains and other genome elements from Ensembl. Regions with pre-existing changes that match those found in BED files of structural variants are reported along with supporting information. We'll discuss practical examples of how this helps improve our ability to utilize structural changes in analysis of tumor variant calls.

The implementation is part of bcbio (https://github.com/chapmanb/bcbio-nextgen), which provides a configuration file and command line interface for running variant analysis on distributed machines. We have an open development community which contributed to our current cancer calling support (http://bcb.io/2015/03/05/cancerval/). We actively develop and support bcbio and hope to grow the community of users who both contribute and use it for answering biological questions.

BOSC 2015 – *Standards and Interoperability*
Day One, 10 July 2015, 14:00 – 15:30
Session chair: Peter Cock

This talk is accompanied by poster #7.

Title	Portable workflow and tool descriptions with the CWL
Authors	Peter Amstutz[0], Nebojša Tijanić[1], Stian Soiland-Reyes[2], John Kern[3], Luka Stojanovic[1], Tim Pierce[0], John Chilton[4], Maxim Mikheev[5], Samuel Lampa[6,7], Hervé Ménager[8], Scott Frazer[9], Venkat Sai Malladi[10], Michael R. Crusoe[11]
Affiliations	0. Curoverse Inc. 1. Seven Bridges Genomics, Inc. 2. University of Manchester, School of Computer Science 3. AccuraGen Inc. 4. Penn State University, The Galaxy Project 5. BioDatomics Inc. 6. Uppsala University, Department of Pharmaceutical Biosciences 7. BILS (Bioinformatics Infrastructure for Life Sciences) 8. Institut Pasteur 9. The Broad Institute 10. Stanford University 11. University of California, Davis, School of Veterinary Medicine
Contact	crusoe@ucdavis.edu
URLs	http://common-workflow-language.github.io/ https://github.com/common-workflow-language/common-workflow-language/
License	Apache License, Version 2.0

Bioinformatics workflow platforms provide provenance tracking, execution and data management, repeatability, and an environment for data exploration and visualization. Example F/OSS bioinformatics workflow platforms include Arvados, Galaxy, Mobyle, iPlant DiscoveryEnvironment, Apache Taverna and Yabi. Each one presently represent workflows using different vocabularies and formats, and adding new tools requires different procedures for each system.

Neither the description of the *workflows* nor the descriptions of the *tools* that power them are usable outside of the platforms they were written for. This results in duplicated effort, reduced reusability, and impedes collaboration.

Three engineers (Peter Amstutz, John Chilton, and Nebojsa Tijanic) from leading bioinformatics platform teams (Curoverse, Galaxy Team, and Seven Bridges Genomics) and a tool author (Michael R. Crusoe / khmer project) started working together at the BOSC 2014 Codefest with an initial focus on developing a portable means of representing, sharing and invoking command line tools which was then the basis for portable workflow descriptions. The group placed high value on re-using existing formats and ontologies; they governed themselves with a lazy consensus / do-ocracy approach.

On March 31st, 2015 the group released their second draft of the Common Workflow Language specification. The serialized form is a YAML document that is validated by an Apache Avro schema and can be interpreted as an RDF graph using JSON-LD. The documents are also valid Wf4Ever 'wfdesc' descriptions after a simple transformation. Future drafts will include the use of the EDAM ontology to describe the tools enabling discovery via the ELIXIR tool registry.

Seven Bridges Genomics, the Galaxy Project, and the organization behind Arvados (Curoverse) have started to implement support for the Common Workflow Language, with interest from other projects and organizations like Apache Taverna, BioDatomics and the Broad Institute. Developers on the Galaxy Team are exploring adding CWL tool description support with plans to add support for the CWL workflow descriptions. Tool authors and other community members will benefit as they will only have to describe their tool and workflow interfaces once. This will enable scientists, researchers and other analysts to share their workflows and pipelines in an interoperable and yet human readable manner.

Title	From peer-reviewed to peer-reproduced: a role for research objects in scholarly publishing
Author	*Alejandra Gonzalez-Beltran* [1], Peter Li [2], Jun Zhao [3], Maria Susana Avila-Garcia [4], Marco Roos [5], Mark Thompson [5], Eelke van der Horst [5], Rajaram Kaliyaperumal [5], Ruibang Luo [6], Tin-Lap Lee [7], Tak-wah Lam [6], Scott C. Edmunds [2], Susanna-Assunta Sansone [1], Philippe Rocca-Serra [1]
Affiliation	[1] Oxford e-Research Centre, University of Oxford, UK [2] GigaScience, BGI HK Research Institute, Hong Kong [3] InfoLab21, Lancaster University, UK [4] Nuffield Department of Medicine, University of Oxford, UK [5] Department of Human Genetics, Leiden University Medical Center, The Netherlands [6] HKU-BGI Bioinformatics Algorithms and Core Technology Research Laboratory & Department of Computer Science, University of Hong Kong, Pokfulam, Hong Kong [7] School of Biomedical Sciences and CUHK-BGI Innovation Institute of Trans-omics, The Chinese University of Hong Kong, Shatin, Hong Kong
Contact URL	alejandra.gonzalezbeltran@oerc.ox.ac.uk http://isa-tools.github.io/soapdenovo2/ https://github.com/ISA-tools/soapdenovo2

The reproducibility of science in the digital age is attracting a lot of attention and concerns from the scientific community, where studies have shown the inability to reproduce results due to a variety of reasons, ranging from unavailability of the data to lack of proper descriptions of the experimental steps.

Multiple research object models have been proposed to describe different aspects of the research process. Investigation/Study/Assay (ISA) is a widely used general-purpose metadata tracking framework with an associated suite of open-source software, which offers a rich description of the experiment's hypotheses and design, investigators involved, experimental factors, protocols applied. The information is organised in a three-level hierarchy where 'Investigation' provides the project context for a 'Study' (a research question), which itself contains one or more 'Assays' (taking analytical measurements and key data processing and analysis steps). Nanopublication (NP) is a research object model which enables specific scientific assertions, such as the conclusions of an experiment, to be annotated with supporting evidence, published and cited. Lastly, the Research Object (RO) is a model that enables the aggregation of the digital resources contributing to findings of computational research, including results, data and software, as citable compound digital objects.

For computational reproducibility, platforms such as Taverna and Galaxy are popular and efficient ways to represent the data analysis steps in the form of reusable workflows, where the data transformations can be specified and executed in an automatic way.

In this presentation, we will address the question of whether such research object models and workflow representation frameworks can be used to assist in the peer review process, by facilitating evaluation of the accuracy of the information provided by scientific articles with respect to their repeatability.

Our case study is based on an article on a genome assembler algorithm published in GigaScience, but due to the proven use of the respective research object models in their respective communities, we argue that the combination of models and workflow system will improve the scholarly publishing process, making science peer-reproduced.

BOSC 2015 – *Standards and Interoperability*
Day One, 10 July 2015, 14:00 – 15:30
Session chair: Peter Cock

Demystifying the Interoperability of Disparate Genomic Resources

Daniel Blankenberg[1, 2] and the Galaxy Team[2]

[1] Department of Biochemistry and Molecular Biology, Penn State University, University Park, PA 16802, USA. Email: dan@bx.psu.edu
[2] http://galaxyproject.org.

Project Website: galaxyproject.org **License**: Academic Free License version 3.0 **Source Code**: github.com/galaxyproject/galaxy

Galaxy (http://galaxyproject.org) is an open, web-based platform for accessible, reproducible, and transparent computational biomedical research. Galaxy makes bioinformatics analyses accessible to users lacking programming experience by enabling them to easily specify parameters for running tools and workflows. Analyses are made transparent by allowing users simple access to share and publish analyses via the web and create Pages, interactive, web-based documents that describe a complete analysis.

Galaxy provides experimental biologist access to powerful analysis infrastructure through the web. Within Galaxy, users are able to upload their own data, graphically execute command-line analysis tools, and interactively build visualizations on their results. In many circumstances, a user is able to conduct their entire analysis without leaving the Galaxy application. However, this is not always the case, as users may find themselves in a situation where they want to utilize data from an external data warehouse or where they may want to continue an analysis at an external resource.

Although Galaxy strives to provide an all-inclusive analysis platform, there are times when external resources provide functionality that is not available within, or is superior to, those embedded directly as a part of Galaxy. These apparent weaknesses actually highlight some of Galaxy's strongest features. Galaxy embraces external resources by providing frameworks for retrieving and sending data seamlessly through its graphical interface. This is more than simply transferring files between two web-servers.

When retrieving data from external resources, file content is only one part of the puzzle. There is often important metadata associated with the datasets, such as genome build identifiers, formats, sample information, column assignments, versions, etc., that are required for the datasets to be useful. Galaxy enables this through the use of Data Source tools, where the framework has been recently enhanced to allow external resources to send multiple files at a time and to provide extensive amounts of metadata. Galaxy currently supports several external data resources, including UCSC table browser, Biomart, InterMine, European Nucleotide Archive, and GenomeSpace. And adding more is a straightforward process.

When sending datasets to an external resource, there are typically two different methods available. In the first method, a standard Galaxy tool can be defined that allows a user to pick datasets from their history, configure any additional options, and then click *Execute* to send the data to the external resource. Data export tools will typically create a new HTML Galaxy History item that contains the results and hyperlinks to allow the user to transition to the external resource. The second method is generally used for enabling external visualization tools. Using the external display application framework, link-outs to external resources, such as UCSC Genome Browser, Integrative Genome Viewer (IGV), and GBrowse, are embedded directly within the dataset preview in the user's history. The user can simply click the link under their dataset and will be forwarded directly to the external application along with their data. In cases when the external display requires different formats or additional index files, such as viewing a VCF file within IGV (i.e. bgziped and indexed with tabix), standard Galaxy converter tools can be automatically utilized by the framework.

Here, we demonstrate a typical NGS analysis where we provide our own data, retrieve data from an external data source, perform an analysis on these data within Galaxy and then send our data to additional external resources for further analysis and visualization. We then explore the basic steps that were needed to enable these external interactions within Galaxy.

Increasing the utility of Galaxy workflows.

John Chilton[1] and The Galaxy Team
[1]Department of Biochemistry and Molecular Biology, Penn State University, University Park, PA 16801, US
jmchilton@bx.psu.edu
Project: http://galaxyproject.org/ Code: http://github.com/galaxyproject/galaxy

Galaxy is a popular data analysis platform that is most often used to integrate diverse command-line utilities into a consistent and intuitive web-based interface. A long standing selling point of Galaxy has been that it allows researchers to extract sample analysis histories out into reusable workflows and build such workflows de novo. Despite the popularity of this feature, the kinds of workflows that could be expressed by Galaxy have had critical limitations. Two of the most glaring of these are that Galaxy workflows have required a fixed number of inputs and the workflow engine planned out every job right at submission time (workflows would cause a series of jobs to queue up in Galaxy - but there was no real workflow engine that could alter the structure of this computation over time). Many relatively basic analyses in bioinformatics require running a variable number of inputs across identical processing steps (“mapping”) and then combining or collecting these results into a merged output (“reducing). Likewise - pausing workflows, splitting inputs up into multiple datasets, and conditionals all require the re-evaluation of workflows overtime. This presentation will discuss how we have started addressing these limitations. In particular we will present dataset collections and a real, pluggable Galaxy workflow subsystem - together these features address the limitations described above and vastly increase the expressiveness of Galaxy workflows.

Galaxy dataset collections are powerful way to group collections into potentially nested hierarchies of lists and pairs of datasets. Existing Galaxy tools can be used without modification to “map” operations across dataset collections to produce new collections with sample information maintained. Likewise tools that consume many datasets can be readily used to “reduce” these collections. For newly developed tools - a wide range of extensions to Galaxy tooling format exist to consume and produce dataset collections. In addition to presenting these additions to Galaxy, extensions to the workflow system to tie together these analyses and innovative UI elements such as the paired list dataset collection builder will be presented.

Specific biologically relevant examples to highlight the power of dataset collections and the new workflow engine will be presented. These will include an RNA-seq workflow based on the tuxedo suite of tools that can process any number of samples and a workflow that exploits the ability to output collections to achieve greater parallelization than was previously possible.

These extensions are powerful new features that greatly enhance the expressivity of Galaxy workflows, but much work remains to do be done. A road map for the future of Galaxy workflows will be laid out - including conditionals, iteration, and more flexible connections between steps (e.g. mapping output metadata to input parameters for instance), etc....

BOSC 2015 – *Standards and Interoperability*
Day One, 10 July 2015, 14:00 – 15:30
Session chair: Peter Cock

Kipper: A software package for sequence database versioning for Galaxy bioinformatics servers

Damion Dooley[1], Aaron Petkau[2], Gary VanDomselaar[2], William Hsiao[1,3]

[1] University of British Columbia, Vancouver, BC, Canada. Email: damion.dooley@bccdc.ca

[2] National Microbiology Laboratory, Public Health Agency of Canada, Winnipeg, MB, Canada.

[3] BC Public Health Microbiology and Reference Laboratory, Vancouver, BC, Canada.

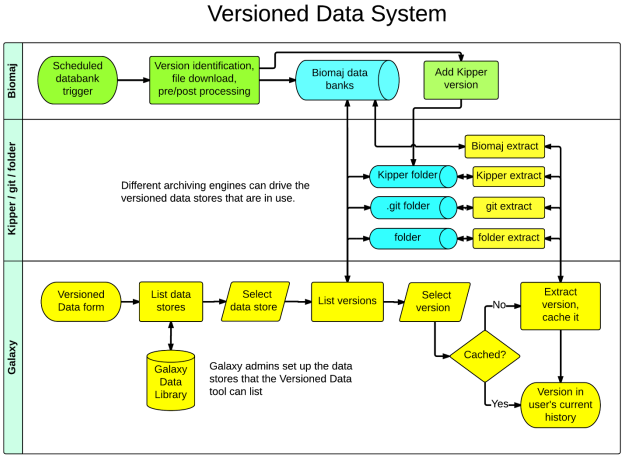
Project Website: https://github.com/Public-Health-Bioinformatics
Source Code: https://github.com/Public-Health-Bioinformatics/versioned_data

Abstract

There are various reasons to rerun bioinformatics tools and pipelines on sequencing data, including re-creating a past result, re-validation of a tool or workflow using a known dataset, or tracking the impact of database changes. For identical results to be achieved, updated reference sequence databases must be versioned. Server administrators have tried to fill the requirements by supplying users with one-off versions of databases, but these are time consuming to set up and to maintain. Disk storage and data backup performance has also discouraged maintaining multiple versions of databases since databases such as NCBI nr can consume 50Gb or more disk space per version, with growth rates that parallel Moore's law.

Our end-to-end open source versioning system combines our own Kipper software package - a simple key-value large-file versioning system - with Biomaj(a software system for downloading sequence databases), and Galaxy (a web-based bioinformatics workflow scheduling platform). Available versions of databases can be recalled and used via command-line or within Galaxy. The Kipper data store format makes publishing curated fasta databases especially convenient since in most cases it can store a range of versions into a file only slightly larger than the size of the latest version.

Kipper is under active development and we encourage feedback from the user community to improve its utility.



BOSC 2015 – *Standards and Interoperability*
Day One, 10 July 2015, 14:00 – 15:30
Session chair: Peter Cock

This talk is accompanied by poster #8.

Evolution of the Galaxy tool ecosystem - *happier developers, happier users*

Martin Čech[1], Björn Grüning[2], Eric Rasche[3], Kyle Ellrott[4], John Chilton[1], and Galaxy Team
[1] Department of Biochemistry and Molecular Biology, PSU, USA marten@bx.psu.edu
[2] Bioinformatics Group, Department of Computer Science, University of Freiburg, Germany
[3] Center for Phage Technology, Texas A&M University, USA
[4] Center for Biomolecular Science & Engineering, University of California Santa Cruz, USA
Project: http://galaxyproject.org Code: https://github.com/galaxyproject License: AFL 3.0

Galaxy is a powerful open-source platform for data intensive research. The number of Galaxy tool developers and users is growing due both to its steady expansion of the number of Galaxy instances into diverse settings (e.g. GalaxyP, deepTools, CoSSci, Oqtans, Genomic Hyperbrowser, Osiris)[1] and initiatives such as the DREAM SMC-Het[2] challenge (which includes a reproducibility focused sub-challenge requiring submission of Galaxy tools and workflows). The result is a need to scale tool testing, discovery, and distribution.

Galaxy uses the Tool Shed as an App Store-like platform for tool exploration and deployment with reproducible workflow sharing support. Today the Tool Shed contains more than 3000 tools from different areas of computational research and a vibrant community is updating and improving these tools.

Driven by community feedback solicited via questionnaire[3] we identified and focused on the areas that would benefit the most from improvements:

- ➔ *Tool testing* - we built Planemo that dramatically simplifies tool creation and testing.
- ➔ *Tool discovery* - we have rewritten search from the ground up to allow Galaxy deployers to more easily identify high quality tools.
- ➔ *Tool distribution* - by providing integration with Jenkins and Github we have simplified and automated the tool testing and Tool Shed publishing process.

A community based commission (IUC) is maintaining a best practice guide to address the first issue and defines standards for high quality Galaxy tools[4].

We also present Planemo - an easy to install command-line utility that gives developers a way to bootstrap, lint, test, and explore Galaxy tools without even requiring installation and configuration of Galaxy. Planemo makes existing developers more productive, and Planemo virtual appliances are being used to introduce new SMC-Het developers to tool and workflow development - lowering the barrier to entry.

Jenkins build scripts leveraging Planemo have been developed to automate testing and deployment of tool repositories, providing vital feedback by posting results back to GitHub. The scripts can also automatically deploy tool repositories to the Tool Shed, reducing the overhead of managing large number of tools.

We firmly believe the presented work will enable the Galaxy community to scale up its already great contributions and boost collaboration.

---

[1] https://wiki.galaxyproject.org/PublicGalaxyServers
[2] https://www.synapse.org/#!Synapse:syn2813581
[3] Data available at https://wiki.galaxyproject.org/Community/GalaxyAdmins/Surveys/2014
[4] http://galaxy-iuc-standards.readthedocs.org/

BOSC 2015 – *Standards and Interoperability*
Day One, 10 July 2015, 14:00 – 15:30
Session chair: Peter Cock

This talk is accompanied by poster #9.

Title	Bionode - Modular and universal bioinformatics
Author	*Bruno Vieira*, Yannick Wurm
Affiliation	http://sbcs.qmul.ac.uk
Contact	mail@bmpvieira.com
URL	http://bionode.io
License	MIT

The exponential growth of biological data generated from sequencing in the last 10 years is putting a lot of pressure in the bioinformatics analysis downstream. However, most tools were developed in an unsustainable way, with a monolithic architecture, complex APIs, custom file formats and without good software development practices. This makes it very hard to build flexible and reproducible analysis pipelines that reuse those tools. In addition, the migration of most utilities to the web has led many developers to rewrite some functionality in JavaScript (a language that can be directly interpreted by browsers).

Bionode aims to solve bioinformatic problems while building highly reusable tools and code, by following the best development practices coming from web startups. Each Bionode module aims to be both a pipeable command line tool that follows the UNIX philosophy (*"try to do just one thing well"*) but also a JavaScript module that can be integrated in web applications or server-side JavaScript (Node.js). This while using standards like JSON file format and Google's Protocol Buffers.

This allows for an unprecedented amount of flexibility in the usability of Bionode. Unlike other similar bio* libraries that require the usage of a specific language, Bionode only requires the usage of JavaScript for web development use cases, and can thus be used via command line in other languages or workflow projects (e.g., Galaxy, NoFlo, and Node-RED). The architecture based on Node.js Streams allows to write complex pipelines that scale without running out of memory by piping chunks of data around.

The high level of modularity of the project also makes it easier for development and contributions, due to the split of functionality into small modules, individually hosted on GitHub/NPM with continuous integration, testing and code coverage. Everything is 100% Open Source (MIT license) and development discussions happen publicly. We also collaborate with Dat (*"git for data"*) and BioJS (*"represent biological data on the web"*).

BOSC 2015 – *Standards and Interoperability*
Day One, 10 July 2015, 14:00 – 15:30
Session chair: Peter Cock

This talk is accompanied by poster #10.

Title	The EDAM Ontology
Author	*Hervé Ménager*, Matúš Kalaš, Jon Ison
Affiliation	Institut Pasteur, University of Bergen, ELIXIR DK
Contact	edam@elixir-dk.org
URL	http://edamontology.org
License	http://edamontology.org/page#License

Bioinformaticians handle an increasingly large and diverse set of tools and data. Meanwhile, researchers demand ever more powerful and convenient means to organise, find, understand, compare, select, use and connect the available resources. These tasks often rely on consistent, machine-understandable descriptions of the underlying components, but these have been generally lacking in ad hoc resource descriptions. The urgent need - filled by EDAM - is for an ontology that unifies semantically the bioinformatics concepts in common use, provides the curator with a comprehensive controlled vocabulary that is broadly applicable, and supports new and powerful search, browse and query functions.

EDAM is an ontology of well established, familiar concepts that are prevalent within bioinformatics, including types of data and data identifiers, data formats, operations and topics. EDAM is a simple ontology - essentially a set of terms with synonyms and definitions - organised into an intuitive hierarchy for convenient use by curators, software developers and end-users.

EDAM is suitable for large-scale semantic annotations and categorization of diverse bioinformatics resources, and also suitable for diverse application including for example within workbenches and workflow-management systems, software distributions, and resource registries.

Version 1.9 of EDAM has been released. Contributions and suggestions are welcome!

BOSC 2015 – *Open Science and Reproducibility*
Day Two, 11 July 2015, 10:45 – 12:30
Session chair: Mónica Muñoz-Torres

Title	A curriculum for teaching Reproducible Computational Science bootcamps
Author	*Hilmar Lapp*, Participants of the Reproducible Science Curriculum Hackathon
Affiliation	Duke University, Center for Genomic and Computational Biology; and National Evolutionary Synthesis Center (NESCent)
Contact	hilmar.lapp@duke.edu
URL	https://github.com/Reproducible-Science-Curriculum/Reproducible-Science-Hackathon-Dec-08-2014
License	Creative Commons Zero

Verifiability and reproducibility are among the cornerstones of the scientific process. They are what allows scientists to "stand on the shoulder of giants". Maintaining reproducibility requires that all data management, analysis, and visualization steps behind the results presented in a paper are documented and available in full detail. Reproducibility here means that someone else should either be able to obtain the same results given all the documented inputs and the published instructions for processing them, or if not, the reasons why should be apparent.

Making the computational components of a research study fully reproducible can be very challenging or even impossible if attempted post-hoc, such as when prompted by journal requirements or funder expectations. Yet, there are techniques, tools, and practices that aleady exist and that stand to help practicing scientists in organizing, documenting, and automating their digital work so that it can be more easily repeated and built upon later by others, including by their future selves. Despite this potential, such tools and practices are rarely taught, and many biological and other domain scientists remain unaware of them.

In this talk we report on an effort, the Reproducible Science Curriculum Hackathon, to develop and teach a 2-day bootcamp-style workshop curriculum for basic techniques, tools, and best practices for life scientists that promote the reproducibility of their computational work from the start. The curriculum is motivated throughout by reproducibility not primarily for the future benefit of others, but for the benefit of accelerating one's own research, for example by more easily repeating and extending analyses with new data, new tools, or parameter modifications. The effort is an international collaboration of a diverse group of people from different fields, disciplines, career tracks and stages, some of whom are actively prototyping and developing tools to improve the reproducibility of data and compute-intensive research. We will present the basic lessons that comprise the curriculum, and results from teaching them at the first two workshops held in May and June 2015. If successful, we expect the workshops to be replicated through train-the-trainer events, much similar to the Software Carpentry and Data Carpentry workshops and teaching model.

BOSC 2015 – *Open Science and Reproducibility*
Day Two, 11 July 2015, 10:45 – 12:30
Session chair: Mónica Muñoz-Torres

This talk is accompanied by poster #11.

Research shared: www.researchobject.org

Matthew Gamble[1], Norman Morrison[1], Stian Soiland-Reyes[1], Carole Goble[1],

[1] School of Computer Science, University of Manchester, Kilburn Building, Oxford Road, Manchester M13 9PL. Email: gamble@cs.manchester.ac.uk

Project Website: http://www.researchobject.org/
Source Code:
https://github.com/apache/incubator-taverna-language/tree/master/taverna-robundle
License: Apache v2
https://github.com/myGrid/ruby-ro-bundle **License**: BSD
https://github.com/gambl/ro-python **License**: MIT

researchobjects.org is a community project that has developed an approach to describe and package up all resources used as part of an investigation as Research Objects (RO's).

RO's - provide two main features; a manifest - a consistent way to provide a well-typed, structured description of the resources used in an investigation; and a 'bundle' - a mechanism for packaging up manifests with resources as a single, publishable unit.

RO's therefore carry the research context of an experiment - data, software, standard operating procedures (SOPs), models etc - and gather together the components of an experiment so that they are findable, accessible, interoperable and reproducible (FAIR). RO's combine software and data into an aggregative data structure consisting of well described reconstructable parts.

RO's have the potential to address a number of challenges pertinent to open research including: a) supporting interoperability between infrastructures by using ROs as a primary mechanism for exchange and publication b) supporting the evolution of research objects as a living collection, enabling provenance tracking c) providing the ability to pivot research object components (data, software, models) that are not restricted to the traditional publication.

Here we present work towards the development and adoption of ROs:

(i) A series of specifications and conventions, using community standards, for the RO manifest and RO bundles.
(ii) Implementations of Java, Python and Ruby APIs and tooling against those specifications;
(iii) Examples of representations of the RO models in various languages (e.g. JSON-LD, RDF, HTML).

BOSC 2015 – *Open Science and Reproducibility*
Day Two, 11 July 2015, 10:45 – 12:30
Session chair: Mónica Muñoz-Torres

This talk is accompanied by poster #12.

Nextflow: a tool for deploying reproducible computational pipelines

Paolo Di Tommaso[1,2], Maria Chatzou[1,2], Pablo Prieto Barja[1,2], Emilio Palumbo[1,2], Cedric Notredame[1,2]

[1] Bioinformatics and Genomics Program, Centre for Genomic Regulation (CRG), Dr. Aiguader 88, 08003 Barcelona, Spain. Email: paolo.ditommaso@crg.eu

[2] Universitat Pompeu Fabra (UPF), 08003 Barcelona, Spain.

Project Website: http://nextflow.io
Source Code: https://github.com/nextflow-io/nextflow
License: GPLv3 - http://www.gnu.org/copyleft/gpl.html

Main Text of Abstract

Genomic pipelines usually rely on a combination of several pieces of third party research software. Academic software applications tend to be prototypes and are often difficult to install, configure and deploy. Furthermore their experimental nature can result in frequent updates, thus raising serious reproducibility issues.

Despite the fact that many tools have been developed to tackle these problems none of them have so far provided a comprehensive solution. For this reason we developed Nextflow, a tool that is specifically designed to address the reproducibility problem in computational pipelines by allowing researchers to easily write parallel and distributed data analysis applications. The three main strengths of Nextflow are:

- Its capacity to integrate any existing tools and scripts.
- Its support of a high-level parallelization model for complex task interactions and easy deployment.
- Increased reproducibility thanks to its reliance on Docker containers technology.

A Nextflow pipeline is made by putting together several processes. Each process can be written in any scripting language that can be executed by the Linux platform (BASH, Perl, Ruby, Python, etc.). Parallelization is automatically managed by the framework and it is implicitly defined by the processes input and output declarations.

Moreover Nextflow provides an abstraction over the underlying execution platform. Thus, the resulting pipeline can run on a single workstation, on different grid infrastructures or in a cloud environment.

The integration with Docker containers technology and the Github sharing platform enables pipelines to be deployed, along with all their dependencies, across multiple platforms without any modifications, making it possible to share them and replicate their results in a predictable manner. In addition Nextflow provides a rich set of built-in functions for recurrent operations on common bioinformatics data formats (FASTA, FASTQ, etc.) such as split, count, filter, combine, etc.

Finally the Nextflow programming model greatly simplifies writing large scalable pipelines, by utilizing a scatter-process-gather parallelization strategy that is quite common in bioinformatics applications.

We used this approach in PIPER, a pipeline for the detection and mapping of long non-coding RNAs. We managed to speed-up the overall pipeline execution by 6 times and to reduce the application code base in a significant manner when compared to the previous PERL based implementation.

BOSC 2015 – *Open Science and Reproducibility*
Day Two, 11 July 2015, 10:45 – 12:30
Session chair: Mónica Muñoz-Torres

This talk is accompanied by poster #13.

Free beer today: how iPlant + Agave + Docker are changing our assumptions about reproducible science

John M. Fonner[1*], Rion Dooley[1], Matthew W. Vaughn[1]

[1] University of Texas at Austin, Austin, TX, USA.

*Email: jfonner@tacc.utexas.edu

Project Website: http://agaveapi.co
Source Code: https://bitbucket.org/taccaci/agave
License: BSD-2Clause (https://bitbucket.org/taccaci/agave/src/6af91b4d4046d7148d448087ac182efc502bc931/LICENSE?at=2.1.0)

Main Text of Abstract

In genomics and other data intensive sciences, collaboration and reproducibility have been severely constrained by three pervasive problems: data availability, software portability, and the onerous computational skillset required to overcome those two things. Today, effective bioinformaticians must be warrior poets that can both grapple with novel experiments in their scientific domain while still expressing their ideas in thousands of lines of scripts and code. While training and educating researchers will always be critical, it is a disservice to scientific discovery to dilute the time of domain experimental researchers with learning an ever-changing computational skillset. The real question is how can we keep scientists focused on their domain research by drastically reducing the overhead in performing responsible, collaborative, data intensive science?

Through the iPlant Collaborative project, we have been building cyberinfrastructure that solves scaling, collaboration, and reproducibility issues while quietly keeping full provenance and versioning information. By hiding these processes behind accessible web interfaces, we are tricking domain researchers into doing the right thing with their data and results, and command line is not required. We invite hackers, thinkers, and makers to peek behind the curtain; to see how Docker containers, the RESTful APIs of Agave, and web user interfaces can work together; and then, since everything is free and open source, to try tweaking things yourselves.

BOSC 2015 – *Open Science and Reproducibility*
Day Two, 11 July 2015, 10:45 – 12:30
Session chair: Mónica Muñoz-Torres

The 500 builds of 300 applications in the HeLmod repository will at least get you started on a full suite of scientific applications

Aaron Kitzmiller[1], John Brunelle[2], Michele Clamp[1], James Cuff[3]

[1] Informatics and Scientific Applications, Faculty of Arts and Sciences, Harvard University. Email: aaron_kitzmiller@harvard.edu

[2] Google, Mountain View, California.

[3] Research Computing, Faculty of Arts and Sciences, Harvard University.

Project Website: http://rc.fas.harvard.edu/helmod
Source Code: https://github.com/fasrc/helmod
License: GPL v2.0 (https://www.gnu.org/licenses/gpl-2.0.html)

Scientific research of any significant scale is fundamentally dependent on free, open source software. While much of this software can be downloaded and run relatively easily on single user systems, many tools have complex, parallel or optimized code that must be compiled *in situ*. Large, shared high performance / high throughput clusters have additional deployment challenges.

The Harvard FAS Odyssey cluster accumulated more than 2000 application and library installs managed by a conventional Linux module system in its first 4 years of existence. These module installations had a number of problems including dependency conflicts and irreproducible build conditions. Migration to the TACC Lmod system alleviates module conflicts, but adds additional burdens and does not solve reproducibility issues.

The Harvard Extensions for Lmod deployment (HeLmod) system enhances a standard rpm-based build tool suite with macros and scripts that automate the build of scientific software for compiler and MPI branches of an Lmod deployment. Builds are captured in rpm spec files, and checked in to a freely available github repository. Over the past year, more than 500 builds have been generated for more than 300 applications. By capturing the requirement for reproducible builds, these spec files demonstrate strategies for many challenging software situations including 1) building for multiple compilers, 2) non-standard library locations, 3) commercial software, 4) scripting interactive builds, and 5) patching broken software and build components. These spec files can either be used directly or as a guide to solving troublesome builds in other systems.

Module dependencies and build metadata are captured in easily parsable text files. These files are used to populate web tools for browsing and search and the generation of dependency lists that enable automated rebuilding against new operating systems, compiler versions, etc.

In addition to spec files and rpms, the system is currently being expanded to provide Dockerfiles and images for general use.

BOSC 2015 – *Open Science and Reproducibility*
Day Two, 11 July 2015, 10:45 – 12:30
Session chair: Mónica Muñoz-Torres

This talk is accompanied by poster #14.

Bioboxes: Standardised bioinformatics tools using Docker containers.

Peter Belmann[1*], Michael Barton[2*], Andreas Bremges[1,3], Johannes Dröge[3], Felipe Leprevost[4], Yasset Perez-Riverol[5], Albert J. Vilella[6], Alex Copeland[2], Alice McHardy[3], Alexander Sczyrba[1]

[1] Faculty of Technology and Center for Biotechnology, Bielefeld University, Germany. Email: pbelmann@cebitec.uni-bielefeld.de

[2] DOE Joint Genome Institute, Walnut Creek, USA.

[3] Helmholtz Centre for Infection Research, Braunschweig, Germany.

[4] Fiocruz, Carlos Chagas Institute, Curitiba, Brazil.

[5] European Molecular Biology Laboratory, European Bioinformatics Institute, Hinxton, Cambridge, United Kingdom.

[6] Onto.it Holdings Ltd., Cambridge, United Kingdom.

[*] Joint First Authors

Project Website: http://bioboxes.org/
Source Code: https://github.com/bioboxes
License: MIT (https://github.com/bioboxes/rfc/blob/master/LICENSE)

We introduce the open-source community "bioboxes" which has the aim of simplifying bioinformatics tools through adopting common interfaces. The Docker project makes installing, running and reproducing the output of an application easier because all the dependencies can be provided by creating a "container" of the application. The bioboxes project furthers this concept by creating an interface standard so that containerised software of the same biobox type can be seamlessly interchanged.

Feedback and contributions of new biobox interfaces are managed using the Github issue system. Bioboxes provides documentation and software to help developers follow this standard. The bioboxes.org website provides instructions and guide on how to build a biobox. We also provide a validator tool that tests whether a container follows a biobox interface and thereby helps the developer create their own bioboxes. Furthermore we provide bioboxes in Github and our public Docker Hub repository for download and use.

Researchers with a whole selection of bioboxes at their fingertips can comprehensively evaluate many similar tools that have the same interface and improve the quality of their research. A system of well-defined bioboxes is an important step towards the making provenance easier and research more shareable and reproducible.

BOSC 2015 – *Open Science and Reproducibility*
Day Two, 11 July 2015, 10:45 – 12:30
Session chair: Mónica Muñoz-Torres

Title	The perfect fit for reproducible interactive research: Galaxy, Docker, IPython
Author	Björn Grüning[1], Eric Rasche[2], John Chilton[3] and Dannon Baker[4]
Affiliation	[1]Bioinformatics Group, Department of Computer Science, University of Freiburg, Germany
	[2]Center for Phage Technology, Texas A&M University, USA
	[3]Department of Biochemistry and Molecular Biology, PSU, USA
	[4]Department of Biology, Johns Hopkins University, USA
Contact	bjoern.gruening@gmail.com
URL	https://github.com/bgruening/docker-galaxy-stable
License	MIT

For years Galaxy has made advanced bioinformatics software accessible to biologists directly by providing an intuitive webinterface to these applications while fostering reproducibility through the automatic creation of re-runnable protocols of each analysis. With the Tool Shed, Galaxy gained a flexible deployment platform enabling identical software installations across Galaxies.

A major hurdle in using Galaxy today is simply finding an instance with the correct set of tools and with enough computational power and storage necessary for a particular analysis. One interesting solution to this challenge is to move the software (Galaxy and the tools) to the data instead of shipping data to a remote server running the correct software; it's also worth noting that many researchers and institutions simply cannot ship their data elsewhere.

Galaxy is using Docker to solve this problem in a way that is even more reproducible by delivering the entire software stack in a container. Each new release of Galaxy is now available as a production-ready Docker container. Additionally, this Docker image can be extended to build personalized Galaxy flavours, with site-specific sets of tools. For example, a Galaxy Docker flavour containing all necessary tools for RNA-seq analysis, or a genome annotation flavour with the NCBI BLAST suite. These flavours are simple to create and can be easily deployed on Linux, OS-X and Windows.

Likewise, Galaxy now allows running tools securely in Docker containers. The process isolation provided by running Galaxy jobs in Docker containers provides a much higher degree of security than running them as native processes, at least in part due to inability to access other users' data. An exciting new development in this Galaxy/Docker ecosystem is the Galaxy-IPython project. IPython is a popular platform providing a web-based interactive computing and visualization environment. Galaxy-IPython allows Galaxy users to run IPython inside Galaxy and access it via their web browser. Additionally, it extends the default IPython environment by providing easy, secure access to Galaxy, it's API, and the user's data. As Galaxy-IPython is deployed on the Galaxy server, it removes the overhead of big-data downloads and uploads during analysis. All of these features work to enable rapid, iterative, and interactive bioinformatics analysis and software prototyping directly in Galaxy, next to your big data.

Galaxy is a popular tool for teaching bioinformatics applications to biologists - Galaxy IPython is a huge step towards enabling it to be a teaching tool for bioinformatics programming as well.

BOSC 2015 – *Open Science and Reproducibility*
Day Two, 11 July 2015, 10:45 – 12:30
Session chair: Mónica Muñoz-Torres

This talk is accompanied by poster #15.

Title	COPO - Bridging the Gap from Data to Publication in Plant Science
Authors	*A Etuk[1]*, F Shaw[1], A Gonzalez-Beltran[2], P Rocca-Serra[2], A Abdul-Rahman[2], P Kersey[3], R Bastow[4], S Sansone[2], R Davey[1]
Affiliations	The Genome Analysis Centre (1), Oxford e-Research Centre, University of Oxford (2), European Bioinformatics Institute (3), University of Warwick (4)
Contact	robert.davey@tgac.ac.uk
URL	https://documentation.tgac.ac.uk/display/COPO/
License	Common Public Attribution License Version 1.0 (CPAL)

The aim of open science is to make scientific research accessible, facilitating experimental reproducibility and transparency. Mechanisms exist for preserving and publishing research objects in plant science within the "omics" fields. However, researchers are often hindered by: (i) complicated and time-consuming procedures for repository deposition; (ii) a lack of interoperability between disparate information sources and mechanisms; (iii) sub-optimal search and retrieval facilities across data repositories; (iv) a lack of public awareness of existing services.

To address these issues, we are developing COPO (Collaborative Open Plant Omics), a brokering service which enables aggregation and publishing of research outputs by plant scientists, and provides access to services across disparate sources of information via web interfaces, and Application Programming Interfaces (APIs) for bioinformaticians. COPO comprises a web front-end (based on the Django framework), data grid infrastructure using iRODS (www.irods.org), and a number of APIs which facilitate the creation and management of logical profiles containing heterogeneous but related research objects representing a body of work. A profile can contain, for example, references to omics and image data, source code, PDF files, or posters and presentations, which are fully described with associated ISA-based metadata. COPO leverages the Investigation/Study/Assay (ISA) formats and ISA software suite tools (http://www.isa-tools.org) to enable experimental metadata attribution and conversion between metadata formats. Based on ISA research objects, this metadata comprises information about the investigators, objectives/hypotheses, related publications, subjects, experimental design and assays. Deposition to public repositories is enabled, providing a technical continuity between services. COPO will provide access to distributed resources through a single point of entry, integrating existing data services such as: European Nucleotide Archive (ENA) (www.ebi.ac.uk/ena) and MetaboLights (www.ebi.ac.uk/metabolights) for raw data; GigaScience for data and workflow citation (www.gigasciencejournal.com); f1000 (www.f1000research.com); Nature Publishing Group Scientific Data (www.nature.com/sdata/) for data publication; figshare (www.figshare.com) for a variety of supplementary data types; Galaxy (www.galaxyproject.org) and iPlant (www.iplantcollaborative.org) for data analysis; GitHub (www.github.com) for source code and ORCID (www.orcid.org) to provide user information. COPO does not store the data itself, but rather references deposited objects, thus keeping the system lightweight and decentralised from third party repositories. DOIs (Digital Object Identifiers) will be minted and mapped to COPO profiles providing permanent digital identities for these collections of research objects.

As research objects deposited through COPO will be accompanied by rich metadata implemented in JSON-LD (www.json-ld.org), cutting edge technologies such as MongoDB (www.mongodb.org) and neo4J (www.neo4j.org) are being used to create a web of linked semantic knowledge, allowing for user customised suggestions for future avenues of investigation. Such inferences are domain agnostic, with the potential for bioinformaticians to discover useful results and techniques from other fields.

We present a prototype front-end with the ability to create user accounts and work profiles. Data transfers to the ENA are possible with associated metadata being input into intuitive online forms. Metadata is stored in a document based database, with transfers between data formats (ISA metadata in RDF and JSON-LD) being handled by ISA software suite tools.

BOSC 2015 – *Open Science and Reproducibility*
Day Two, 11 July 2015, 10:45 – 12:30
Session chair: Mónica Muñoz-Torres

This talk is accompanied by poster #16.

ELIXIR UK building on Data and Software Carpentry to address the challenges in computational training for life scientists

Aleksandra Pawlik[1], Aleksandra Nenadić[1], Susanna-Assunta Sansone[2], Alejandra Gonzalez-Beltran[2], Carole Goble[1]

[1] University of Manchester, UK; [2] University of Oxford, UK

Email: aleksandra.pawlik@manchester.ac.uk

Project Website: http://elixir-uk.org/

Within life sciences training in lab skills for scientific computing and data manipulation faces several challenges. The discoverability of training material is often poor. High assumptions about prior knowledge render it effectively unusable. As a result, many scientists do not even attempt to use them. Others use training material for self-teaching without support or guidance from experienced instructors.

To address these challenges ELIXIR UK leveraged the work of the existing Software Carpentry and Data Carpentry initiatives and launched them for ELIXIR Europe in order to train researchers in core computational and data manipulation skills. ELIXIR UK is one of the nodes constituting pan-European ELIXIR project for supporting and developing a sustainable infrastructure for life science research, and its translation outside of academia. Software and Data Carpentry provide training materials and workshop for computational skills in science.

The main goals of the of ELIXIR UK working with Software and Data Carpentry are:

- Adopt the training model of Data and Software Carpentry to suit the needs of the life science research communities within ELIXIR,
- Collaboratively develop new and extend existing training materials tailoring them to the various life sciences areas,
- Build the self-sustained ELIXIR-wide network of certified instructors though "train the trainer" programme and workshops,
- Expand the training network of organisations within ELIXIR able to run independently Software and Data Carpentry workshops.

To support an effective kick-off of these activities ELIXIR granted funding for a pilot project "Working up and building the foundation for Data Carpentry and Software Carpentry within ELIXIR". The pilot consists of three events: a collocated training materials development hackathon and a Data Carpentry workshop (the first one took place in March 2015, hosted by the Finnish node, and the second one will be in June, hosted by the Dutch node) and a "train the trainer" workshop to be hosted in early 2016 by the Swiss node. The attendance at the events of the representatives of as many nodes as possible helped with the outreach and scaling up the whole project. The hackathons provided an excellent opportunity for introducing in depth the node representatives to the Carpentries' teaching model, approach and training materials. The events were run by two experienced Software and Data Carpentry certified instructors (one of them is also a member of the Steering Committees for both initiatives). This helped in discussions at the events, clarifying and answering all questions that the node representatives had. The points made by the ELIXIR participants were fed back to the leads of Software and Data Carpentry leads. This two-way communication was an opportunity to understand better how the Carpentry model can be used beyond the original workshops and serve the life sciences research community.

BOSC 2015 – *Open Science and Reproducibility*
Day Two, 11 July 2015, 10:45 – 12:30
Session chair: Mónica Muñoz-Torres

This talk is accompanied by poster #17.

Parallel recipes: towards a common coordination language for scientific workflow management systems

Yves Vandriessche

Software Languages Lab, Vrije Universiteit Brussel, Brussels, Belgium. Email: yvdriess@vub.ac.be
Exascience Life Lab, Imec, Leuven, Belgium

Project Website: http://www.exascience.com/
Source Code: http://github.com/yvdriess/precipes/
License: BSD three-clause license

It is evident from examining best practices pipeline implementations that there is significant inherent complexity: coordinating the execution of a large number of heterogeneous applications, informally specified data formats, heavy dependence on execution context (e.g. environment variables, specific Java VM version), etc. Glue languages such as Perl or bash are great tools for handling this type of complexity. However, the traditional glue languages were not made to deal with challenges faced in today's distributed computing environments. Using a simple software lines of code metric on standard sequencing pipelines has shown us an increase from roughly 250sLoC to 2500sLoC, an order of magnitude increase. Scientific workflow management systems have stepped in and have greatly simplifying the task of gluing together analysis jobs into a larger reusable whole. However, as a recent report on parallelisation in such systems shows[1]: *"In closing, while substantial progress has been made in parallel scientific workflow enactment, the field of solutions is still heterogeneous and leaves room for improvement."*

We report on the an initial experiment where we tackle the coordination challenges inherent in the large scale distribution and parallelisation of bioinformatics workflows. We work towards a common coordination language with which workflow management systems can coordinate with the various actors involved in the execution: analysis tools, operating system, other workflow management systems, schedulers, compute infrastructures, etc. The rationale behind such common coordination language follows the basic claims put forward by Gelernter and Carriero[2] of Linda[3] fame: that it is possible to treat coordination as *orthogonal* to computation and that it is possible to define coordination in a *general* way such that it applies to every asynchronous part of the system. Concretely, we implemented a simplified workflow description language called *parallel recipes* or *precipes*. As a case study we implemented the standard exome sequencing pipeline in *precipes*. Important is how this workflow's parallel execution is implemented using the Concurrent Collections (CnC) coordination language model[4]. We use the Intel CnC++ implementation (https://icnc.github.io/) as an execution platform and execute transparently on top of a workstation, cluster or Amazon EC2 nodes. We demonstrate automatic exploitation of in-node as well as across-node parallelisation with predictable linear scaling.

---

[1] M. Bux and U. Leser, "Parallelization in Scientific Workflow Management Systems," CORD Conference Proceedings, 2013.

[2] D. Gelernter and N. Carriero, "Coordination languages and their significance," Commun. ACM, vol. 35, no. 2, p. 96, 1992.

[3] D. Gelernter, "Generative communication in Linda," ACM Transactions on Programming Languages and Systems (TOPLAS), vol. 7, no. 1, pp. 80–112, 1985.

[4] Z. Budimlić, M. Burke, V. Cavé, K. Knobe, G. Lowney, R. Newton, J. Palsberg, D. Peixotto, V. Sarkar, and F. Schlimbach, "Concurrent collections," Scientific Computing, vol. 18, no. 3, pp. 203–217, 2010.

BOSC 2015 – *Open Science and Reproducibility*
Day Two, 11 July 2015, 10:45 – 12:30
Session chair: Mónica Muñoz-Torres

This talk is accompanied by poster #18.

openSNP - personal genomics and the public domain

Bastian Greshake[1], Philipp Bayer[2], Helge Rausch, Julia Reda

[1]Department for Applied Bioinformatics, Goethe University, Frankfurt am Main, Germany Email: bgreshake@googlemail.com

[2] School of Plant Biology, University of Western Australia, Perth, Australia

Project Website: https://openSNP.org
Source Code: https://github.com/gedankenstuecke/snpr
License: Code: MIT, Data: Creative Commons Zero

Direct-To-Consumer (DTC) genetic testing is still a rather recent but growing phenomenon, with some companies now having as many as 850k paying customers (1). DTC genetic tests are still largely based on the analysis of Single Nucleotide Polymorphisms (SNPs) with micro arrays. The analysis of such SNP data sets in general, using Genome-Wide Association Studies (GWAS), has lead to the discovery of many genotype-phenotype associations for a wide array of phenotypic traits and diseases (2,3). The growing number of customers also enabled providers of DTC genetic testing to perform their own GWAS (4).

By and large the data generated in the field of DTC genetic testing is not openly available to third parties, like academic researchers, citizen scientists, and hobbyist genealogists. This is partially due to privacy and ethical concerns (5,6), but also due to financial interests of the DTC providers (7,8). With ***openSNP*** we created an easily accessible platform to enable customers of DTC to dedicate their own DTC genetic testing data along with their phenotypes into the public domain using Creative Commons Zero. Additionally the platform offers annotations for those SNPs generated by mining different public and open data sets, such as the *Public Library of Science*, the *SNPedia* (9), *Mendeley* and more.

Since *openSNP* started at the end of 2011 people have used it to release over 1700 genotyping files into the public domain and included information for over 289 phenotypes, thus creating a free and growing resource for scientists and citizen scientists alike. With this the project actively contributes to the discussion on open human genetic data, such as bioethical implications and privacy research (10,11), genealogy, teaching, pharmacogenomics (12) and even art (13).

REFERENCES

(1) http://mediacenter.23andme.com/blog/2015/02/19/fdabloomupdate/
(2) Johnson A, O'Donnell C. BMC Med. Genet. (2009); 10: 6.
(3) Hindorff LA, Sethupathy P, Junkins HA, Ramos EM, Mehta JP, et al. Proc Nat Acad Sci USA. (2009); 106: 9362-9367.
(4) Do CB, Tung JY, Dorfman E, Kiefer AK, et al. PLoS Genet. (2011); 7(6): e1002141.
(5) Caulfield T, McGuire AL. Annu. Rev. Med. (2011); 63: 1.1-1.11.
(6) Hogarth S, Javitt G, Melzer D. Annu. Rev. Genomics Hum. Genet. (2008); 9: 161–82.
(7) http://mediacenter.23andme.com/blog/2015/01/06/23andme-genentech-pd/
(8) http://mediacenter.23andme.com/blog/2015/01/12/23andme-pfizer-research-platform/
(9) Cariaso M and Lennon G, Nucl. Acids Res. (2012) 40 (D1):D1308-D1312.
(10) Angrist M, PLoS ONE (2014) 9(3): e92060.
(11) http://seclab.soic.indiana.edu/GenomePrivacy/papers/Genome%20Privacy-paper4.pdf
(12) Samwald et al. BMC Medical Informatics and Decision Making (2015) 15:12
(13) https://soundcloud.com/thesoundofpeople/the-sound-of-bastian-greshake-6-channels-remixed

BOSC 2015 – *Translational Bioinformatics*
Day Two, 11 July 2015, 14:00 – 14:40
Session chair: Brad Chapman

This talk is accompanied by poster #19.

CIViC: Crowdsourcing the Clinical Interpretation of Variants in Cancer

Malachi Griffith[1], Obi L Griffith[2], Nicholas C Spies[3], Kilannin C Krysiak[4], Benjamin J Ainscough[5], Adam C Coffman[6], Josh F McMichael[7], James M Eldred[8], Dave E Larson[9], Jason R Walker[10], Elaine R Mardis[11], Richard K Wilson[12]

[1] The Genome Institute, Washington University School of Medicine, St. Louis, MO. Email: mgriffit@genome.wustl.edu

[2-12] The Genome Institute, Washington University School of Medicine, St. Louis, MO.

Project Website: https://civic.genome.wustl.edu (or www.civicdb.org)
Mission Statement: https://civic.genome.wustl.edu/#/collaborate

Source Code:
https://github.com/genome/civic-server
https://github.com/genome/civic-client

The landscape of the genomics of tumorigenesis has been systematically surveyed in recent years, identifying thousands of potential cancer-driving alterations. However, few resources exist to facilitate prioritization and interpretation of these alterations in a clinical context. Interpreting the events from even a single case requires both extensive bioinformatics expertise as well as an understanding of cancer biology and clinical paradigms. Genomic aberrations must be placed in the context of therapeutic response and diagnostic or prognostic associations. The evidence for these associations must be captured and characterized so that we can achieve a principled consensus among genomic experts, pathologists, and oncologists on how best to interpret a genomic alteration in a clinical context. This interpretation step now represents a significant bottleneck, preventing the realization of personalized medicine. To this end, we present CIViC (www.civicdb.org) as a forum for the clinical interpretation of variants in cancer. We believe that to succeed, such a resource must be comprehensive, current, community-based and above all, open-access. CIViC allows curation of structured evidence coupled with free-form discussion for user-friendly interpretation of clinical actionability of genomic alterations. CIViC supports multiple lines of evidence, stratified based on the type of study, from *in vitro* studies to large clinical trials. CIViC currently contains clinical interpretations for over 135 genomic alterations in 55 genes spanning 45 cancer types and summarizing the evidence from nearly 230 publications. The CIViC interface facilitates both discovery and collaboration by allowing a user to not only search and browse the current state-of-the art interpretations but also to join the community discussion by adding, editing, or commenting on genomic events, evidence for their clinical actionability and the resulting community consensus interpretation.

BOSC 2015 – *Translational Bioinformatics*
Day Two, 11 July 2015, 14:00 – 14:40
Session chair: Brad Chapman

This talk is accompanied by poster #20.

From Fastq To Drug Recommendation – Automated Cancer Report Generation using OncoRep & Omics Pipe

Tobias Meissner[1,3*], Kathleen M. Fisch[2,3*], Louis Gioia[3] & Andrew I. Su[3]

[1] Department of Molecular and Experimental Medicine, Avera Cancer Institute, La Jolla, CA, USA. Email: tobias.meissner@avera.org

[2]Department of Medicine, Center for Computational Biology & Bioinformatics, University of California, San Diego, La Jolla, CA, USA.

[3] Department of Molecular and Experimental Medicine, The Scripps Research Institute, La Jolla, CA, USA.

* both authors contributed equally to this work

Project Website: http://sulab.org/tools/oncorep-oncogenomics-report/
Source Code: https://bitbucket.org/sulab/oncorep/ & https://bitbucket.org/sulab/omics_pipe/
License: MIT

Abstract

Next generation sequencing allows us to study cancer in a large scale and in multiple dimensions. This provides in-depth insight into tumor pathogenesis on the individual patient level, paving the path to precision medicine. The advent of precision medicine introduces a shift in how cancer patients will be treated in the future, away from the one drug-one disease paradigm towards the idea of bringing the right drug to the right patient.

Challenges that arise with this paradigm shift are i) High-throughput, automated, parallel, and reproducible processing and analysis of sequencing data in the n-of-1 setting, ii) extracting, integrating, interpreting and reporting relevant information from various layers of omics data iii) integrating omics data with drug databases, (iv) presenting the information in an understandable and timely manner to provide clinically relevant and actionable targets to the clinician and tumor board, and (v) creating a transparent, open-source, community framework for enabling community curation of best practices and algorithms, and reproducible analysis pipelines in a move towards standardizing next generation sequencing analysis for patient care.

To address these challenges, we present an automated cancer reporting framework based on two open-source platforms, OncoRep and Omics Pipe. OncoRep is an RNAseq-based n-of-1 reporting tool for cancer patients that has been integrated into Omics Pipe, a community-based framework that enables reproducibility by curating and automating best practice omics data analysis pipelines. We have applied this framework to breast cancer patients in an n-of-1 setting. RNA-seq was performed for each patient and the fastq files were processed and analyzed using Omics Pipe, generating expression data, variants and fusions based on published methods. OncoRep was incorporated as a custom pipeline in Omics Pipe to perform prospective molecular classification, detect altered genes and pathways, identify gene fusion events and clinically actionable mutations, and to report suitable drugs based on identified actionable targets. Omics Pipe automates these analyses and keeps a detailed record of the analyses and parameters used to ensure reproducibility. OncoRep integrates and visualizes the data in an approachable html based interactive report as well as a PDF based summary report, providing the clinician and tumor board with an approachable report to guide the treatment decision making process. We hope with our contribution to enhance transparency and reproducibility of next generation sequencing analysis and clinical reporting, which will be needed for moving this technology into routine clinical practice.

BOSC 2015 – *Translational Bioinformatics*
Day Two, 11 July 2015, 14:00 – 14:40
Session chair: Brad Chapman

This talk is accompanied by poster #21.

Cancer Informatics Collaboration and Computation: Two Initiatives of the U.S. National Cancer Institute

Authors: Ishwar Chandramouliswaran[1], Juli D. Klemm[1], Tanja Davidsen[1], Anthony R. Kerlavage[1], Warren Kibbe[1]
Author affiliations: [1]National Cancer Institute, Center for Biomedical Informatics and Information Technology
Presenting author email address: Ishwar.chadramouliswaran@nih.gov
URL for project website: www.nciphub.org,
http://cbiit.nci.nih.gov/ncip/nci-cancer-genomics-cloud-pilots/nci-cloud-initiative
Code URL:
Open Source License:

One of the objectives of the National Cancer Institute's Center for Biomedical Informatics and Information Technology is to focus on innovative programs and technologies to support the use of informatics in cancer research and encourage open science. We have several initiatives underway that are designed to enhance community driven software development, democratize access to cancer data, tools & standards as well encourage innovative approaches that improve access to software applications & build scientific communities that enable deep collaboration around specific research questions.

At BOSC 2013, we presented NCI's open development initiative to share open-source code for cancer informatics software applications through GitHub at https://github.com/ncip. This presentation describes two follow-on initiatives and how these initiatives can help researchers engaged in cancer informatics collaborate and contribute to NCI's open science efforts.

NCIP Hub: One of our goals is to enable cancer researchers to create community driven, adaptive, and collaborative environments that promote the exchange of research ideas and resources. To address this we have established the NCIP Hub, using the open source HUBzero Platform for Scientific Collaboration, to provide this online collaboratory for the cancer informatics community. The intent is to empower community members to both contribute and use software tools, data, standards, or other relevant digital assets to an ever-growing research and educational resource. The work products become discoverable and citable, and their impact can be measured. We hope that this will contribute to the creation of a 'community impact score' based on data sharing, algorithm sharing, software sharing, discoverability, annotation, and of course use and reuse. Individuals engaged in cancer informatics can become a member and contribute at www.nciphub.org.

The Cancer Genomics Cloud Pilots: The purpose of the Cancer Genomics Cloud Pilots is to support the development of a new model for computational analysis of biological data that can address the required computational capacity for storage, analysis, and discovery. In this new model the data repository is co-located with computational capacity that can be accessed either via a web interface or via an Application Programming Interface (API) while ensuring data security. Developers of analytical software applications will also be able to bring new tools to the data, and researchers will be able to bring their own data to the cloud to analyze in the context of TCGA data. The Cancer Genomics Cloud has the potential to democratize access to NCI-generated genomic data and provide a less resource intensive and more cost-effective way to perform computational analysis for the cancer research community while enabling reproducibility of the analyses. The three pilot systems being developed by the Broad Institute, the Institute for Systems Biology and Seven Bridges Genomics are expected to be available for community use and evaluation in early 2016. Stay informed by following http://cbiit.nci.nih.gov/ncip/nci-cancer-genomics-cloud-pilots/nci-cloud-initiative.

BOSC 2015 – *Project Updates*
Day Two, 11 July 2015, 14:40 – 15:30
Session chair: Nomi Harris

Biopython Project Update 2015

João Rodrigues[*], Tiago Antão [†], Peter Cock[‡], Wibowo Arindrarto[§], Michiel de Hoon[¶], Eric Talevich[‖], and the Biopython Contributors

16th Bioinformatics Open Source Conference (BOSC) 2015, Dublin, Ireland

Website: http://biopython.org
Repository: https://github.com/biopython/biopython
License: Biopython License Agreement (MIT style, see http://www.biopython.org/DIST/LICENSE)

The Biopython Project is a long-running distributed collaborative effort, supported by the Open Bioinformatics Foundation, which develops a freely available Python library for biological computation [1].

We present here details of the latest Biopython release - version 1.65. New features include: extended Bio.KEGG and Bio.Graphics modules to support the KEGG REST API, as well as parsing, representing, and drawing KGML pathways; inclusion of the new NCBI genetic code table 24 (Pterobranchia Mitochondrial) and corresponding translation functionality in Bio.Data; improvements to Bio.SeqIO (parse and index_db methods) and Bio.SearchIO (hit retrieval using alternative IDs); and a rewritten Bio.SeqUtils.MeltingTemp with additional methods to calculate oligonucleotide melting temperatures. Additionally, we continued our efforts to abide by the PEP8 coding style guidelines, namely using lowercase module names in new experimental modules.

We are currently preparing a new release - version 1.66 - that will feature additional improvements to the Bio.KEGG and Bio.Graphics modules (support for transparency in KGML pathways), extended support for the "abi" format in Bio.SeqIO, miscellaneous improvements to the test suite, and further adherence to PEP8. In addition, our participation in Google Summer of Code 2014 had Evan Parker adding lazy-parsing support for Bio.SeqIO. The additions are currently under review and should soon be integrated.

Finally, complementary to these developments, we created a new repository for Docker containers. The included containers support both Python versions 2 & 3 and install all of Biopython's dependencies. They are, therefore, useful for development, but also for teaching due to the inclusion of IPython Notebooks.

References

[1] Cock, P.J.A., Antao, T., Chang, J.T., Chapman, B.A., Cox, C.J., Dalke, A., Friedberg, I., Hamelryck, T., Kauff, F., Wilczynski, B., de Hoon, M.J. (2009) Biopython: freely available Python tools for computational molecular biology and bioinformatics. *Bioinformatics* **25**(11) 1422-3. doi:10.1093/bioinformatics/btp163

[*]Bijvoet Center for Biomolecular Research, Utrecht University, Utrecht, NL. Email: j.rodrigues@uu.nl
[†]Vector Biology Department, Liverpool School of Tropical Medicine, Pembroke Place, UK
[‡]Information and Computational Sciences, James Hutton Institute (formerly SCRI), Invergowrie, Dundee, UK
[§]Sequencing Analysis Support Core, Leiden University Medical Center, Leiden, NL
[¶]Division of Genomic Technologies, RIKEN Center for Life Science Technologies, Yokohama, JP
[‖]Department of Dermatology, University of California San Francisco, San Francisco, CA, USA

BOSC 2015 – *Project Updates*
Day Two, 11 July 2015, 14:40 – 15:30
Session chair: Nomi Harris

The biogems community: Challenges in distributed software development in bioinformatics

George Githinji[1,10], Ben Woodcroft[2,10], Joachim Baran[3,10], Francesco Strozzi[4,10], Raoul Bonnal[5,10], Naohisa Goto[6,10], Toshiaki Katayama[7,10], Hiroyuki Mishima[8,10], and Pjotr Prins[9,10]

[1]KEMRI-Wellcome Trust Research Programme, Kenya. **Email:** ggithinji@kemri-wellcome.org
[2]The Australian Centre for Ecogenomics, School of Chemistry and Molecular Biosciences, The University of Queensland, Australia [3]National Evolutionary Synthesis Center, Durham, United States of America. [4]Bioinformatics Core Facility, Parco Tecnologico Padano, Italy. [5]Integrative Biology Program, Istituto Nazionale Genetica Molecolare, Milan 20122, Italy, [6]Department of Genome Informatics, Genome Information Research Center, Research Institute for Microbial Diseases, Osaka University, Suita, Osaka, Japan, [7]Human Genome Center, Institute of Medical Science, University of Tokyo, Tokyo 108-0071, Japan, [8]Department of Human Genetics, Nagasaki University Graduate School of Biomedical Sciences, Nagasaki, Japan, [9]University Medical Centre, Utrecht, Netherlands, [10]The Bioruby Project

Affiliation: The BioRuby Project
Contact E-mail: bioruby@lists.open-bio.org
URL: http://biogems.info/
Source code: Linked from biogems website
License: All licenses are of type approved by the Free Software Foundation (FSF)

The BioRuby project is a comprehensive bioinformatics library for the Ruby programming language. In its more than fifteen years of existence it has evolved from a monolithic to a distributed code-base.

With over 40 contributors and 134 biogems and a plan to enlist other Bio* projects, it is important to assess potential challenges of the distributed contributions model such as testing, documentation, coding standards and communication.

In this talk we present metrics on test coverage, documentation and communication within the biogems projects. We also highlight successful distributed software development models and patterns that encourage new contributions, collaboration and build trust among developers.

BOSC 2015 – *Project Updates*
Day Two, 11 July 2015, 14:40 – 15:30
Session chair: Nomi Harris

This talk is accompanied by poster #22.

Title	Apache Taverna: Sustaining research software at the Apache Software Foundation
Authors	*Stian Soiland-Reyes*, Ian Dunlop, Alan R Williams, Apache Taverna team
Affiliation	Apache Software Foundation; eScience Lab, University of Manchester
Contact	http://taverna.incubator.apache.org/contact
URL	http://taverna.incubator.apache.org/
License	Apache License, Version 2.0

Apache Taverna is a *scientific workflow* system, encompassing a graphical workbench, command line tools, server and APIs. Although Taverna was conceived for bioinformatics, its user base also encompasses domains such as astronomy, digital preservation, biodiversity and virtual physiology. Taverna has been an open source project since 2003, developed by the myGrid consortium and originally led by the University of Manchester and EMBL-EBI. In October 2014, Taverna became an incubating project at the Apache Software Foundation.

Here we describe our motivation for changing the *governance* and *ownership* of the Taverna project, and reflect on our experience and challenges in transitioning a University-led research software activity to an open development process and building a wider developer community.

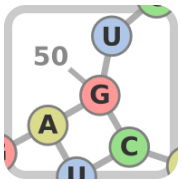
Research Software supports scientists and researchers; its development is usually funded for domain-specific projects and made publicly available as Open Source through code repositories (GitHub, BitBucket). By the time research software develops into a mature code base and gains a diverse user base, the initial funding and related projects may already be finished, yet ownership and control of the project typically remains with the original authors. Users and third-party developers have the *source code*, but are often not included in project decisions, and may not feel *ownership* to contribute code, documentation or support for others.

The original developers may eventually become involved in new projects that do not directly relate to the original project, and may lose focus as the user base changes. Thus it becomes critical to grow a sustainable and diverse *developer community*, and to build an *open governance* model that encourages engagement and commitment from everyone using the software. Efforts like the Software Sustainability Institute are crucial, as it helps guide research software engineers in best practices for making their research software open and maintainable. Ultimately the success of an open source project should lead to a change of its structure and management to widen its developer base.

For Taverna, we considered several options to reduce the lead role and responsibility that the University of Manchester had, and to move to a neutral ownership model where any interested developers could contribute to Taverna's development on an equal standing. One of the options was to create a Taverna Foundation, but this would have required legal administration and a dedicated budget. Another option was to assign copyright and management of the code to a well-established software foundation like the Software Freedom Conservancy, the GNU project at the Free Software Foundation, the Eclipse Foundation, the Outercurve Foundation which is backed by Microsoft, and the Apache Software Foundation. The different options come with implications for the project's way of working, licensing, community, politics, infrastructure and public impression - which must be evaluated for the particular research software project.

By choosing Apache as Taverna's new home, we put emphasis on the community building, a strong and neutral governance model, and clear intellectual property management, which we believe makes the project more approachable for new participants and commercial entities. While the initial months as an Apache Incubator involved a fair bit of administrative overhead, such as moving mailing lists, transitioning web sites and bug trackers, checking dependency licensing and verifying intellectual property ownership of the donated source code, we feel this effort is offset already by an increase in awareness and engagement in the community at large. Several new developers have been attracted to the project, with many proposals and new ideas, including four Google Summer of Code 2015 student applications.

Title	Simple, Shareable, Online RNA Secondary Structure Diagrams
Author	*Peter Kerpedjiev*, Stefan Hammer, Ivo Hofacker
Affiliation	http://www.tbi.univie.ac.at
Contact	pkerp@tbi.univie.ac.at
URL	http://rna.tbi.univie.ac.at/forna
License	GPLv2



The visualization of RNA secondary structure is essential for describing its function. It depicts the inter-molecular pairing patterns that characterize its structure and provide hints as to the role that it might play within the cell. It provides a context for explaining experimental results. A mutation leading to a phenotypic variation can, in some cases, be described in terms of the change to the secondary structure that it induced. Conversely, analysis of the secondary structure diagram can provide hints about where to mutate a sequence in order to test a hypothesis about its biological role. Finally, RNA molecules can be categorized and classified according to their structure, the visualization of which gives researchers and practitioners a visual identifier for a particular class of RNA (most notably, for example, tRNAs and miRNAs).

The actual generation of an RNA secondary structure diagram, however, is not trivial and a number of techniques have been developed to generate visually pleasing and relevant layouts [1, 2, 3, 4, 5]. What most lack, however, is an easily accessible online interface to allow researchers to effortlessly generate beautiful, interactive, customizable diagrams. Our software, dubbed `forna`, is an online application that allows for dead-simple generation of secondary structure diagrams. It allows users to interactively manipulate the layout of the diagram, using a force-directed layout reminiscent jViz.RNA [4], but refined and made accessible online without dependencies.

In addition to providing an interactive, dependency-free, Javascript-based online secondary structure viewer, our software implements a number of features which make it easy for researchers to tailor their diagrams to maximize their expressive capacity. This includes interactively editing the underlying structure (rather than just the layout), overlaying coloring information to express relevant supplementary information such as probing data or conservation. More importantly, we provide a Javascript API to include our visualization container on any web page, allowing researchers to seamlessly share interactive secondary structure diagrams without having to externally generate them or requiring the user to download or allow external dependencies (such as Java) to view them. The result is a portable, reusable and accessible application for converting text-based secondary structure information to a descriptive visual representation which can easily be shared and disseminated.

[1] Bruccoleri,R.E. and Heinrich,G. (1988) An improved algorithm for nucleic acid secondary structure display. *Computer applications in the biosciences: CABIOS,* **4** (1), 167–173.

[2] Byun,Y. and Han,K. (2009) PseudoViewer3: generating planar drawings of large-scale RNA structures with pseudoknots. *Bioinformatics,* **25** (11), 1435-1437.

[3] Darty,K., *et al.* (2009) VARNA: interactive drawing and editing of the RNA secondary structure. *Bioinformatics,* **25** (15), 1974.

[4] Wiese,K.C. *et al.* (2005) jviz. rna-a java tool for RNA secondary structure visualization. *NanoBioscience, IEEE Transactions on,* **4** (3), 212–218.

[5] Hecker,N., *et al.* (2013) RNA secondary structure diagrams for very large molecules: RNAfdl. *Bioinformatics,* **29** (22), 2941–2942.

This talk is accompanied by poster #23.

BioJS 2.0: an open source standard for biological visualization

Tatyana Goldberg[1,2], Guy Yachdav[1,2,3], Sebastian Wilzbach[1], David Dao[1], Iris Shih[1], Michiel Helvensteijn[4], Rafael Jimenez[5], Seth J Carbon[6], Alex García[7], Leyla Garcia[5], Suzanna E Lewis[6], Ian Mulvany[8], Francis Rowland[5], Gustavo Salazar[9], Fabian Schreiber[5,10], Ian Sillitoe[11], Anil Thanki[12], José M Villaveces[13], Henning Hermjakob[5], Burkhard Rost[1,2], Manuel Corpas[12]

[1] TUM, Department of Informatics, Bioinformatics & Computational Biology, 5748 Garching, Germany. Email: gyachdav@rostlab.org

[2] TUM Graduate School, CeDoSIA, 85748 Garching, Germany.

[3] Biosof LLC, New York, NY, 10001, USA.

[4] Leiden Institute of Advanced Computer Science, University of Leiden, 2311 EZ Leiden, Netherlands.

[5] ELIXIR, Hinxton, CB10 1SD, UK.

[6] Lawrence Berkeley National Laboratory, Berkeley, CA, 94720, USA.

[7] School of Library and Information Science, Florida State University, Tallahassee, FL, 32306, USA.

[8] eLife, Cambridge, CB2 1JP, UK.

[9] Computational Biology Group, University of Cape Town, 7925 Cape Town, South Africa.

[10] The Wellcome Trust Sanger Institute, Hinxton, Cambridge, CB10 1SD, UK.

[11] Biomolecular Structure and Modelling Group Department of Biochemistry, University College London, London, WC1E 6BT, UK.

[12] The Genome Analysis Centre, Norwich Research Park, Norwich, NR4 7UH, UK.

[13] Max Planck Institute of Biochemistry, 82152 Martinsried, Germany.

Project Website: http://biojs.net/
Source Code: https://github.com/biojs/biojs
License: Apache license V 2.0

BioJS 2.0 is an open-source, JavaScript-based biological data visualization framework. The development of BioJS has been prompted by the growing need for bioinformatics visualization tools to be easily shared, reused and discovered. BioJS 2.0 features an open, inclusive framework for integration of visualization code (called components) that can be combined to create rich and interactive applications. Components can be quickly discovered on the BioJS 2.0 registry (http://biojs.io), a searchable and highly descriptive repository that highlights a component's usefulness by using a star rating system and a download counter. Applying modern best practices, BioJS 2.0 makes it easy for the end users to release and deploy new components. This in turn allows users with no or little technical skills to be able to test and reuse. Among the 90 currently published (and ready to be used) components, the BioJS 2.0 library already incorporates a set of commonly used components such as a Cytoscape network viewer, a 3D molecule viewer, a multiple sequence alignment viewer, a heat map viewer, a phylogenetic tree viewer and a BAM file viewer, applicable to a range of data visualization needs in the –omics fields. Some notable bioinformatics resources that have already integrated BioJS components include UniProt, ENSEMBL, PredictProtein, ApiNATOMY and the Galaxy project. The BioJS community of open source developers is heavily engaged in the organization of workshops, tutorials, hackathons and participates as a mentoring organization in the Google Summer of Code program. The BioJS registry, source code repository, extensive tutorials and documentation can be accessed through http://biojs.net/.

BOSC 2015 – *Visualization*
Day Two, 11 July 2015, 16:00 – 16:30
Session chair: Karsten Hokamp

This talk is accompanied by poster #24.

Visualising Open PHACTS linked data with widgets

Ian Dunlop[1], Carole Goble[2]

[1] University of Manchester, UK. Email: ian.dunlop@manchester.ac.uk

[2] University of Manchester, UK. Email: carole.goble@manchester.ac.uk

Project Website: http://openphacts.org
Source Code: https://github.com/openphacts/openphacts-vis-compoundinfo
License: MIT

The **Open PHACTS Discovery Platform** was developed to reduce barriers to drug discovery in industry, academia and for small businesses. It is based on a linked data API that sits on top of various life sciences datasets eg ChEMBL and Uniprot, see http://dev.openphacts.org. Many users of the API stated that they would like to view the information available within their own web pages but didn't have the time to integrate it. To enable this integration we developed a simple youtube style embedding widget that allows users to very simply embed visualisations of the data in their own environments. The visualisations are based on those available in the Open PHACTS Explorer https://explorer2.openphacts.org. The original widget code is available from https://github.com/openphacts/ops-html-widgets with a live example at http://openphacts.github.io/ops-html-widgets/ . Users can embed the visualisations in their own web pages by including the widget code and adding a simple HTML element to their pages:

```
<div class="compound-info" data-ops-uri="http://www.conceptwiki.org/concept/dd758846-
1dac-4f0d-a329-06af9a7fa413" style="display: none;"></div>
```

This would display the info for "Aspirin", see http://dev.mygrid.org.uk/blog/2014/05/coding-without-coding-announcing-the-open-phacts-html-widgets/. Each widget can be customised using some simple markup that uses the handlebars syntax, see http://handlebarsjs.com/. In the following example only the preferred label, SMILES and InCHI are displayed.

```
<div class="compound-info" data-ops-uri="http://www.conceptwiki.org/concept/dd758846-
1dac-4f0d-a329-06af9a7fa413" style="display: none;">
  <div>Preferred Label: {{prefLabel}}</div>
  <div>SMILES: {{smiles}}</div>
  <div>Inchi: {{inchi}}</div>
</div>
```

The widgets were then redeveloped for the BioJS initiative, see http://biojs.net/ and more recently refactored to use NodeJS, see https://github.com/openphacts/openphacts-vis-compoundinfo. We have also been investigating the use of the emerging web components standard, see http://webcomponents.org/.

BOSC 2015 – *Late-Breaking Lightning Talks*
Day Two, 11 July 2015, 16:30 – 17:00
Session chair: Hilmar Lapp

This talk is accompanied by poster #25.

Biospectra-by-sequencing genetic analysis platform

Patrick Biggs[1], Rudiger Brauning[2], Mingshu Cao[2], Charles David[3], Marcus Davy[3], Helge Dzierzon[3], Rob Elshire[2], Ruy Jauregui[2], Aurelie Laugraud[2*], John McCallum[3], Alan McCulloch[2], Roger Moraga[2], Louis Ranjard[1], Roy Storey[3], Shane Sturrock[1]

[1] NZGL, New Zealand.

[2] AgResearch, New Zealand.

[3] Plant and Food, New Zealand.

*Aurelie.laugraud@agresearch.co.nz

All authors contributed equally to the work and are listed in alphabetical order.

Project Website: http://www.biospectrabysequencing.org/
Source Code: https://github.com/biospectrabysequencing/
License: The GNU Affero General Public License (http://www.gnu.org/licenses/)

Main Text of Abstract

The Biospectra-by-sequencing (BBS) project started in 2014. It is a collaborative multi-institutional effort to build a robust genetic analysis platform. It initially focused on open-source Genotyping-by-sequencing (GBS) technology. GBS reduces genome complexity by using restriction enzymes, making genotyping-by-sequencing cost effective. BBS goes beyond genotyping and extends the application of the reduced complexity sequencing to many other areas. Most importantly, our project aims to encourage a diverse range of end-users to adopt this technology.

Advances in next generation sequencing have consistently increased value for money thereby opening many new opportunities. However, the lack of appropriate analysis pipelines prevents scientists from taking full advantage of these. To realize this potential, we are developing a set of analytical platforms built around GBS. The first aim is to use proof-of-concept work to estimate their applicability across a wide range of biological systems. The second is to promote end-users to adopt these platforms through structured engagement.

The project brings together developers and users from different institutions in New Zealand. We work collaboratively using GitHub as our primary common platform. The core of the analysis engine uses the readily available TASSEL suite, developed at Cornell University. TASSEL takes sequence reads, maps them against a reference genome and then calls variants. We want to incorporate TASSEL in a more automated pipeline with emphasis on reproducible research. The frameworks we use ensure that the same analysis can be redone accurately and/or passed on, thereby facilitating better quality science. We also add pre- and post-processing components. The main focus of pre-processing is quality control.The post-processing part involves filtering after the variant calling. The multidisciplinary background of people in our group provides expertise in most BBS applications and ensures that our software can be used to study all organisms in a large range of experiments.

BOSC 2015 – *Late-Breaking Lightning Talks*
Day Two, 11 July 2015, 16:30 – 17:00
Session chair: Hilmar Lapp

Title	PhyloToAST: Bioinformatics tools for species-level analysis and visualization of complex microbial communities
Author	*Shareef M. Dabdoub* and Purnima S. Kumar
Affiliation	The Ohio State University College of Dentistry Columbus, OH, USA
Contact	dabdoub.2@osu.edu
URL	http://phylotoast.readthedocs.org/
License	MIT

Purpose: Understanding human-associated microbial ecology is essential for insight into health, as well as identifying disease states, risk factors, and etiology. The 16S ribosomal RNA gene is the most common genetic marker for taxonomic identification due to its near universal presence and static function over time, and a wide variety of tools exist for quantifying samples. The popular QIIME software was developed to gather such tools into a single, usable pipeline, but is less usable when species-level analysis is important; as is the case with highly complex oral biofilms. We have developed a new set of tools, called PhyloToAST, that wholly integrate with the QIIME pipeline to reduce primer bias, enhance species-level analysis and visualization, and greatly improve analysis speed.

Methods: All tools were developed using the Python programming language, and use QIIME output files as input, but do not directly depend on a QIIME installation. All analysis and generation of data for visualization was performed with resources provided by the Ohio Supercomputer Center.

Results: Our new pipeline was applied to three oral microbiome datasets examining subgingival bacterial community composition in smokers and non-smokers, diabetics, and dental implant recipients. Compared with the QIIME-provided Amazon EC2 instance, our pipeline reduced processing time for 2 million 16S sequences from over one week to a matter of hours. Furthermore, we developed and applied algorithms to reduce single-primer bias, condense redundant taxonomic output, and automatically generate visual representations of phylogenetic quantification and comparison.

Conclusion: We have developed a set of analysis tools (PhyloToAST) targeted at enhancing taxonomic identification of microbial samples while using the QIIME software pipeline. When applied to several large samples of the oral microbiome, analysis times were drastically decreased and species-level analysis was substantially enhanced. Our tools integrate with the QIIME pipeline and are available free and open source (MIT license).

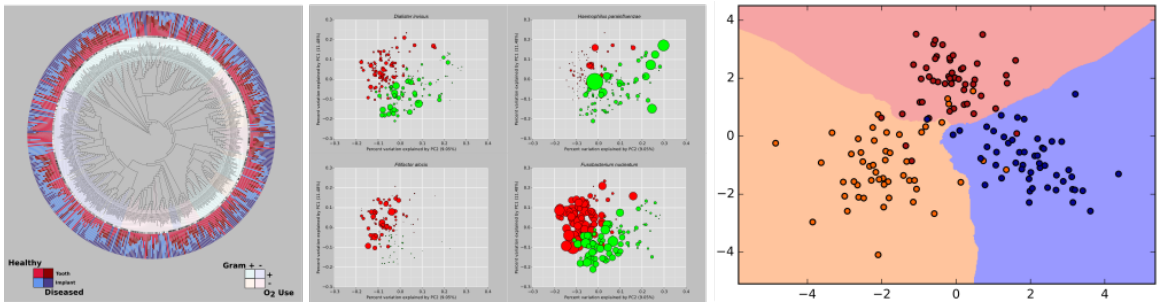


Figure 1: Examples of visualizations created with PhyloToAST. From left to right: A phylogenetic tree with stacked bar charts providing species-level abundance comparisons between four groups. Next is a series of PCoA plots, based on UNIFRAC distances, visually representing the abundance of several species between smokers and non-smokers. Finally, a k-Nearest Neighbors classification of microbial composition data. The input into the KNN algorithm was high-dimensional OTU data simplified with Linear Discriminant Analysis.

BOSC 2015 – *Late-Breaking Lightning Talks*
Day Two, 11 July 2015, 16:30 – 17:00
Session chair: Hilmar Lapp

This talk is accompanied by poster #26.

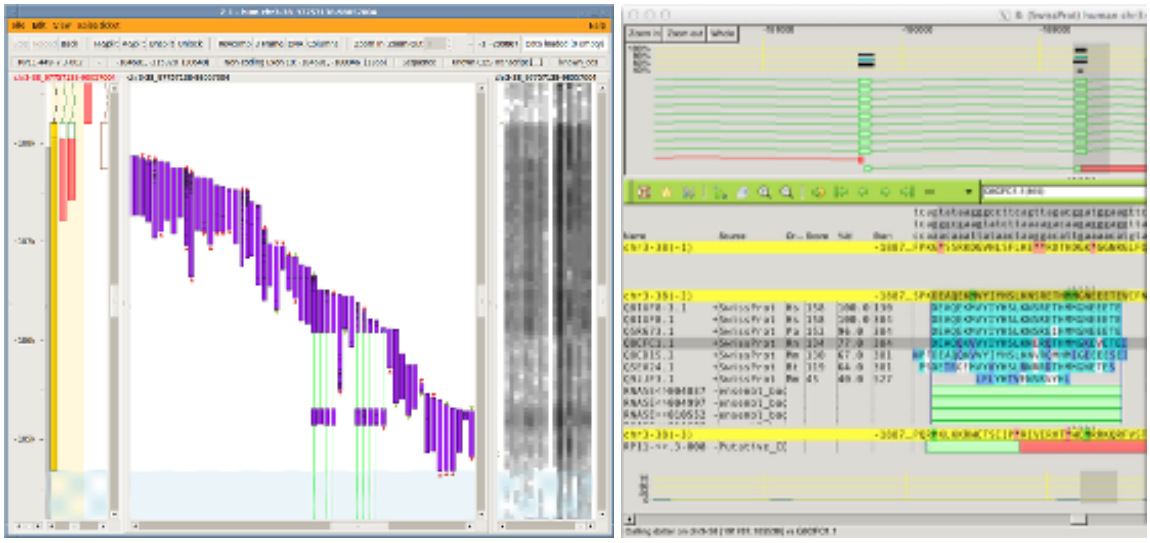
Title	Otter/ZMap/SeqTools: A productive alternative to web browser genome visualisation
Author	*Gemma Guest*, Michael Gray
Affiliation	Wellcome Trust Sanger Institute (http://www.sanger.ac.uk)
Contact	annosoft@sanger.ac.uk
URL	http://www.sanger.ac.uk/resources/software/annosoft/
License	GNU General Public License (GPL)

We believe our set of integrated software tools offers our annotators a genome visualisation and annotation environment that cannot be matched in a browser application. The software's features have evolved over many years to support complex vertebrate annotation using a workflow which pulls together genomic data from around the world.

Otter coordinates the annotation workflow, manages local editing session caches, and records and audits annotation edits. Otter performs on-the-fly alignments to allow annotators to verify new and remapped evidence. Otter's filters can re-map data sources between different sequence assemblies.

ZMap is a high-performance standalone genome browser/editor written in C/C++. Its vertical columns allow for side-by-side comparison of many tens of data sources, drawn from local and remote resources in standard formats including GFF3 and BAM. ZMap allows split panels to be scrolled in unison, and has a rich set of highlighting, searching and filtering options for coping with large-volume alignment and feature sources. ZMap also provides a "remote control" interface that allows another program (e.g. otter) to exchange information with it, so that it can be integrated into existing software.

Otter and ZMap launch Blixem, an interactive browser which provides display of features from ZMap in the context of one-to-many pairwise alignments at individual amino acid or nucleotide base level. Blixem is supported by Dotter for graphical dot-matrix comparison of sequences and by Belvu for viewing of multiple sequence alignments and phylogenetic trees. Blixem, Dotter and Belvu can also be run standalone using standard file formats such as GFF3, FASTA, Stockholm etc.



BOSC 2015 – *Late-Breaking Lightning Talks*
Day Two, 11 July 2015, 16:30 – 17:00
Session chair: Hilmar Lapp

This talk is accompanied by poster #27.

aRchive: enabling reproducibility of Bioconductor package versions

Nitesh Turaga[1], Enis Afgan[1], Eric Rasche[2], Dannon Baker[1] and The Galaxy Team

[1]Johns Hopkins University, Baltimore, MD, USA. Email: nturaga1@jhu.edu
[2] Center for phage Technology, TAMU, College Station, TX, USA.

Project Website: bioaRchive.github.io
Source Code: https://github.com/bioarchive/aRchive_source_code
License: MIT

The Bioconductor suite provides bioinformatics tools in the form of R packages, which have frequent version upgrades. Once an upgrade takes place in a Bioconductor package, it is hard to retrieve previous versions from the source repository. One of Galaxy's primary goals is enabling the reproducibility of any analysis, without the user having to consider it. A major component enabling this is the Galaxy Tool Shed, which provides a host of tools and dependencies that can be installed in Galaxy instances to provide precise versions of tools to Galaxy users. The inability to retrieve specific previous versions of Bioconductor packages makes reproducibility of Bioconductor-based analysis difficult, if not impossible, in Galaxy (or elsewhere). Integrating support for multiple versions of Bioconductor packages within Galaxy would yield immediate improvement for reproducibility while using Bioconductor packages wrapped as Galaxy tools. To that end, we have implemented a method that provides this level of reproducibility for Bioconductor tools in the context of the Galaxy Tool Shed. To do this, we started with a copy of the publicly available, read-only subversion repository of all Bioconductor packages. We then traced through the commit history of each package, extracting released versions and stored them as independent documents. All the versions of all the packages have now been made available in a newly formed, public *aRchive* from where they can easily be retrieved. The *aRchive* is automatically maintained via a cron job that updates the repository with new package versions as they are released. This makes it possible for developers to easily obtain any version of a Bioconductor package required by a pipeline, and make the tool available in the Tool Shed. The *aRchive* ensures previously performed analyses can be reproduced down to the exact version of the initial software used. Thinking to the future, the *aRchive* could be integrated into Bioconductor itself via the *biocLite()* function, with an additional argument specifying a version number of the package. This talk will describe implementation details, results, and future work related to *aRchive* and how it bridges the gap between Bioconductor and Galaxy.

BOSC 2015 – *Late-Breaking Lightning Talks*
Day Two, 11 July 2015, 16:30 – 17:00
Session chair: Hilmar Lapp

This talk is accompanied by poster #28.

Developing an Arvados BWA-GATK pipeline

Pjotr Prins[1,2], Joep de Ligt[3], Isaac Nijman[1], Ward Vandewege[4], Bryan Cosca[4] and Peter Amstutz[4]

[1] University Medical Center Utrecht, Utrecht, The Netherlands
[2] University of Tennessee Health Science Center, Memphis, USA
[3] Hubrecht Institute, Utrecht, The Netherlands
[4] Curoverse, Boston, USA[†]

Contact: pjotr.public05@thebird.nl

Project website: `http://arvados.org/`
Source code: `https://arvados.org/projects/arvados/repository`
License: AGPLv3 for core, Apache 2.0 for the SDK

Arvados is a free and open source platform for bioinformatics and big data science. In this (lightning) talk, we present the porting of an existing HiSeq BWA-GATK based variant calling pipeline from Sun Grid Engine (SGE) to Arvados, resulting in a pipeline that is faster, scalable, simpler and more robust.

The main reason the pipeline runs faster is that the Arvados file system (named Keep) is decentralized. In most bioinformatics cluster environments, the central file storage is the bottleneck because of resource contention, i.e. many cluster nodes are hitting the storage for data requests. When uploading a file into Keep the file is chunked and distributed across nodes. When a node requires data, it can fetch it from different locations in the network.

Because of Keep, the new pipeline is scalable and runs in flat time, i.e. processing one genome takes the same amount of time as running ten or a hundred. Running pipelines in parallel is now only a function of adding new nodes. Keep is distributed and scales accordingly.

The original SGE pipeline consisted of hundreds of lines of Perl and bash code, which mostly dealt with the plumbing of submitting jobs, checking conditions and job completion. In contrast, the Arvados version is mostly single lines for command line invocation as Arvados takes care of the plumbing.

We also needed to speed up GATK using GATK/Queue. Queue chunks BAM files and fans work out to multiple nodes. For this, a special Queue adapter had to be written for Arvados similar to the one that exists for SGE. The Arvadas implementation is an improvement over the SGE implementation because it avoids using mounted NFS for sharing and collating Queue results, thereby making GATK/Queue more robust.

[†]Disclosure: Curoverse is a major contributor to the Arvados open source project and a sponsor of BOSC 2015.

BOSC 2015 – *Late-Breaking Lightning Talks*
Day Two, 11 July 2015, 16:30 – 17:00
Session chair: Hilmar Lapp

This talk is accompanied by poster #29.

Out of the box cloud solution for Next-Generation Sequencing analysis

F. van Dijk[1*], H. Byelas[1*], L. Jensma[2], P. Neerincx[1], D. van Enckevort[1], M. Swertz[1]

[1] Genomics Coordination Center, Department of Genetics, University Medical Center Groningen, The Netherlands; Emails: f.van.dijk02@umcg.nl m.a.swertz@rug.nl

[2] University of Groningen, The Netherlands

Project Website: http://www.molgenis.org/wiki/ComputeVM
Source Code: https://github.com/molgenis/; https://github.com/molgenis/molgenis-pipelines; https://github.com/molgenis/molgenis-compute
License: GNU LESSER GENERAL PUBLIC LICENSE

The Genomics Coordination Center (Groningen, the Netherlands) has gained broad experience in running complex workflows, in which large datasets are analyzed using heterogeneous computational resources from projects like the Genome of the Netherlands [1] and LifeLines. The primary goal was to analyze large data and deliver results as quick as possible.

Now, sharing analysis protocols between institutions and reproducing analysis results has become an important issue. Setting up an execution environment in a new cluster or grid computational site introduces certain latency to the time needed to obtain results. Furthermore, the executional settings environments can change in computational grids and clusters. Hence, we have considered using the cloud infrastructure to automate setting up and sharing analysis infrastructure.

In this work, we present the complete execution environment for NGS analysis, which can be used out of the box in computational resources based on OpenStack platform. We created a configurable VM with widely used in bioinformatics software, implemented into a single pipeline to analyze NGS data. It starts with BWA aligning the raw data (FASTq), followed by realignment around known indels, quality score recalibration and variant calling using Genome Analysis ToolKit (GATK). Several quality metrics are obtained using picard-tools during data processing, variant calls are annotated using SnpEff and SnpSift tools to aid variant interpretation, producing a tab-delimited variant file. We use the EasyBuild framework and wget tool to install software and download resources respectively to ensure reproducibility of environment set-up. This enables users to reproduce installation using pre-defined easyblocks, which are available in the software repository. Furthermore, the necessary resources are installed by executing a shell script after the VM is initiated. The pipeline is available in the github repository.

To summarize, we have shown how to ensure reproducibility and efficient sharing of NGS analyses with the prepared OpenStack VM, which contains all open-source software needed for analysis grouped in a single analysis pipeline. This OpenStack VM can also be combined with MOLGENIS database [2] management system via OpenStack API to track executions. Also, all NGS analysis jobs can be generated as scripts from the MOLGENIS-Compute command-line tool [3]. Scripts can be started manually or via the pre-installed SLURM scheduler.

[1] The Genome of the Netherlands Consortium. Whole-genome sequence variation, population structure and demographic history of the Dutch population. *Nat Genet* 2014; 46:818–825.
[2] Swertz M. *et. al*. The molgenis toolkit: rapid prototyping of biosoftware at the push of a button. *BMC bioinformatics* 2010; 11(Suppl 12)
[3] Byelas H. *et. al*. Scaling bio-analyses from computational clusters to grids. *in proc. of the 5th IWSG conference)*, CEUR-WS.org

[*] Contributed equally.

Poster #30.

Aequatus: Visualising complex similarity relationships among species

Anil S. Thanki[1], Sarah Ayling[1], Javier Herrero[1, 2], Robert P. Davey[1]

[1] The Genome Analysis Centre, Norwich Research Park, Norwich NR4 7UH, UK

[2] Research Dept. of Cancer Biology, UCL Cancer Institute, 72 Huntley Street, London WC1E 6DD

Project Website: http://browser.tgac.ac.uk/aequatus
Source Code: https://github.com/tgac/aequatus-browser
User guide: http://browser.tgac.ac.uk/aequatus-user-guide/
License: GPL V3
Contact: anil.thanki@tgac.ac.uk

The phylogenetic information inferred from the study of homologous genes helps us to understand the evolution of gene families, which plays a vital role in finding ancestral gene duplication events as well as identifying regions those are under positive selection within species (1). Conservation of homologous loci results in syntenic blocks, and there are various tools available to visualise syntenic information between species, such as Ensembl Browser (2), Genomicus (3), SyMap (4), and MizBee (5). These tools are able to provide an overview of syntenic regions as a whole, reaching down to the gene level, but none provide any information about structural changes within genes such as the conservation of ancestral exon boundaries amongst multiple genomes. To this end, we present the Aequatus Browser, a web-based tool with novel rendering approaches to visualise homologous, gene structures among differing species or subtypes of a common species.

The Aequatus Browser utilises common open source web technologies to provide a fast and intuitive browsing experience over complex data, processing and visualising comparative genomics information directly from the Ensembl Compara and Ensembl Core schema databases. Precalculated genomic alignments, in the form of CIGAR strings, are held in Ensembl Compara and Aequatus cross-references these sequences to Ensembl Core databases for each species to gather genomic feature information. Aequatus then processes the comparative and feature data to provide a visual representation of the phylogenetic and structural relationships among the set of chosen species. Whilst applicable to species with high-quality gold-standard reference genomes such as human or mouse, the Aequatus Browser was designed with large fragmented genome references in mind, particularly hard-to-assemble polyploid plants. The ultimate goal of the Aequatus Browser is to provide a unique and informative way to render and explore complex relationships between genes from various species at a level that has so far been unrealised.

The latest version of the Aequatus Browser supports the Ensembl Compara schema v78 and later, as well as refactored code for faster data retrieval, improved visualisation algorithms, and a simplified and informative user interface. It also includes a new REST API for consistent access to genes of interest, making it easy to share information with collaborators via persistent URLs.

References

1. http://www.ncbi.nlm.nih.gov/pubmed/19029536
2. http://www.ensembl.org/
3. http://www.genomicus.biologie.ens.fr/genomicus-75.02/
4. http://www.agcol.arizona.edu/software/symap/
5. http://www.cs.utah.edu/~miriah/mizbee/Overview.html

BOSC 2015 – *Poster-only Abstracts*
Both Days, 10 & 11 July 2015, 13:00 – 14:00

Poster #31.

MOLGENIS Workbench for Systems Medicine

K. Joeri van der Velde[1,2], Mark de Haan[1,2], Cisca Wijmenga[2], Richard J. Sinke[2], Tom J. de Koning[2], Rolf H. Sijmons[2], MOLGENIS team[1,2] and Morris A. Swertz[1,2]

[1] University of Groningen, University Medical Center Groningen, Genomics Coordination Center, Groningen, The Netherlands
[2] University of Groningen, University Medical Center Groningen, Department of Genetics, Groningen, The Netherlands
Contact: k.j.van.der.velde@umcg.nl, m.a.swertz@rug.nl
Project website: http://www.molgenis.org, source code: https://github.com/molgenis

In 2009 we first presented MOLGENIS at BOSC as software generator for complex life science data. Since then, we developed MOLGENIS to tackle more challenges such as high-throughput analysis pipelines (BOSC 2011). This year we present a fresh and modernized MOLGENIS with a data model that now can be fully changed at runtime and includes easier upload format, data explorer, REST/R APIs, visualization and annotation tools with the focus on systems medicine based research and clinical applications.

High-throughput use cases such as multi-omics integration and NGS variant interpretation can now benefit from MOLGENIS adaptable upload formats and query performance, but also require a pre-filled toolbox to help process and understand these data. Therefore we also added extensible variant 'annotators' that enables easy data enrichment (CADD, FitCon, 1000G, ExAC, ClinVar, CGD, HPO, etc.), application analysis protocols (risk prediction, monogenic diagnostic analysis, etc.) and supporting algorithms (discover de-novo variants, symptom-to-disease matching, genome build liftover, etc.). These annotators are also available as a command-line executable to enable use in routine analysis pipelines before uploading the results. Adding more annotators is currently done by implementing a minimal Java interface class. Visual inspection of genes and variants in a biological context is made possible by a WikiPathway-based viewer and Dalliance-powered genome browser.

MOLGENIS is a collaborative open source platform on a mission since 2002 to generate great software infrastructure for life science research. It has already produced a large variety of applications including patient registries, model organism databases, biobank catalogs and computational script generators. We have refreshed the MOLGENIS platform by moving from generation-time to run-time configuration, allowing the users to upload complete data structures, incorporating popular software tools like Maven, MySQL, SpringMVC, GitHub, Bootstrap, Java 8 and ElasticSearch. The resulting modular software suite generates rich web applications that feature an import wizard for flexible formats, APIs for REST and JSON, user and rights management, cross-dataset ontological harmonization, and of course data exploration tools including plotting, filtering, aggregation, complex queries, and metadata browsing. Many components have runtime extension points, meaning custom R plots and reporting templates in Freemarker can be defined and used to present data. Imported data is indexed using ElasticSearch to eliminate long loading times.

We expect the MOLGENIS community will continue to develop valuable Systems Medicine exploration apps as well as function as a sharing platform for best practice data and pipelines, integration with international sharing platforms such as GA4GH and Cafe Variome (for which pilots are underway), well-curated reference knowledge-bases, and optimal user interfaces, results of which can disseminate into research institutes, clinical software companies and individual labs.

Poster #32.

SPINGO: a rapid species-classifier for microbial amplicon sequences

Guy Allard[1], Feargal J Ryan[1], Ian B Jeffery[1] and Marcus J Claesson[1]

[1] School of Microbiology and Alimentary Pharmabiotic Center, University College Cork, Cork Ireland.
Email: feargalr@gmail.com

Project Website: https://github.com/GuyAllard/SPINGO
Source Code: https://github.com/GuyAllard/SPINGO
License: GNU General Public License (https://www.gnu.org/copyleft/gpl.html)

Taxonomic classification is a corner stone for the characterisation and comparison of microbial communities. Currently, most existing methods are either slow, restricted to specific communities, highly sensitive to taxonomic inconsistencies, or limited to genus level classification. These weaknesses mean that these methods may not uncover crucial microbiota information that can be obtained through a high-resolution analysis of the data. It is therefore imperative to increase taxonomic resolution to species level. In response to this need we developed SPINGO, a flexible and stand-alone software dedicated to high-resolution assignment of sequences to species level using 16S rRNA gene regions from any environment. SPINGO compares favourably to other methods in terms of classification accuracy, and is as fast or faster than available tools that have higher error rates. We also demonstrated its flexibility by successfully applying SPINGO to cpn60 amplicon sequences, demonstrating its ability to identify other types of target genes. SPINGO is an accurate, flexible and fast technique for taxonomic assignment down to the species level. This combination is important for the rapid and accurate processing of ever larger amplicon datasets generated by high-throughput next generation sequencing technologies.

BOSC 2015 – *Poster-only Abstracts*
Both Days, 10 & 11 July 2015, 13:00 – 14:00

Poster #33.

Title	ANNOgesic - A computational pipeline for RNA-Seq based transcriptome annotations of bacteria
Author	*Sung-Huan Yu*, Konrad U. Förstner, Jörg Vogel
Affiliation	Institute for Molecular Infection Biology (IMIB), University of Würzburg, Germany
Contact	konrad.foerstner@uni-wuerzburg.de
URL	https://github.com/Sung-Huan/ANNOgesic
License	ISC license

High-troughput RNA sequencing (RNA-Seq) has become a powerful tool to improve the transcriptome/genome annotations of organisms. This technology has helped to detect new transcripts including numerous ones of non-protein-conding genes which are hard to predict purely on the genome sequence. Still, the translation from RNA-Seq data into meaningful annotations is a labor intensive task and lacks streamlining. Here we present the open-source licensed (ISC license) command line tool *ANNOgesic* which provides several subcommands that assist in the RNA-Seq data based generation of high-resolution transcriptome annotations with a focus on bacterial species.

Depending on the specific task the tool requires different input files like the reference genome sequence, RNA-Seq read alignments and available annotations of the organism to study or closely related species. *ANNOgesic* searches for new loci and redefines gene boundaries of those and previously known ones based on transcript assemblies as well as on transcriptional start sites and terminator predictions. It integrates those findings with further information from public sources (like gene functions classifications from Gene Ontology) and produces high-resolution annotations in GFF3 format (Gene feature format). Additionally, it can group genes into operons and suboperons, detect circular RNAs, Single Nucleotide Variation (SNV) as well as processing sites and performs target predictions for newly found sRNAs.

Taken together, *ANNOgesic* offers several functionalities that improve the quality and increase the speed of the transcriptome annotation process significantly.

BOSC 2015 – *Poster-only Abstracts*
Both Days, 10 & 11 July 2015, 13:00 – 14:00

Poster #34.

BioXSD — a data model for sequences, alignments, features, measured and inferred values

Matúš Kalaš[1], Sveinung Gundersen[2], László Kaján[3], Jon Ison[4], Steve Pettifer[5], Christophe Blanchet[6], Rodrigo Lopez[7], Kristoffer Rapacki[4] and Inge Jonassen[1]

[1] Computational Biology Unit, Department of Informatics, University of Bergen, Bergen, Norway.
Email: matus.kalas@uib.no
[2] Department of Informatics, University of Oslo, Oslo, Norway.
[3] unaffiliated (previously Bioinformatics and Computational Biology Department, Technische Universität München, Garching, Germany).
[4] Center for Biological Sequence Analysis, Department of Systems Biology, Technical University of Denmark, Lyngby, Denmark.
[5] School of Computer Science, University of Manchester, Manchester, UK.
[6] French Institute of Bioinformatics, Gif-sur-Yvette, France.
[7] European Bioinformatics Institute, EMBL, Hinxton, UK.

Project Website: http://bioxsd.org
Source Code: https://github.com/bioxsd/bioxsd

Note: We have been exploring ways to adopt, adapt, or develop a license – or a combination of a license and some additional technical and ethical rules - that would be suitable for community-developed, "open" standards for interoperability. While openness for contributions, improvements, and certain kinds of customizations is one of the main goals, another main goal is keeping a standard "standardized" enough to serve the desired interoperability. We would like to work together with O|B|F on establishing such foundations suitable for interoperability standards developed in a participatory and transparent community spirit, and draw attention to **open development and licensing of standards for interoperability** during the BOSC Codefest and the BOSC 2015 itself.

BioXSD has been developed as a universal data model and exchange format for basic bioinformatics types of data: sequences, alignments, features and related values, inferred or measured. The BioXSD data model is rich enough to enable loss-less capture of diverse data that would otherwise require use of multiple different formats, and often even introduction of new formats for untypical features, classifications, or measured values. In BioXSD, an innovatively broad range of experimental data, annotations, and alignments can be recorded in an integrated chunk of data, together with provenance metadata, documentation, and semantic annotation with concepts from ontologies of user's choice.

BioXSD has so far been released in form of a machine-understandable XML Schema (XSD). Ongoing developments concentrate on providing BioXSD in form of JSON Schema and XML Schema 1.1, which may in the future be supplemented by RelaxNG, or even OWL and other data-modelling languages or frameworks. This will enable using BioXSD as a common data model supporting serialization of bioinformatics data into XML, JSON, RDF, or binary (EXI and BSON) as desired, while maintaining consistent and smooth validation, conversions, and parsing into objects for programming. The semantics of BioXSD is defined via SAWSDL references to EDAM (http://edamontology.org) and to a few main Semantic-Web vocabularies.

BOSC 2015 – *Poster-only Abstracts*
Both Days, 10 & 11 July 2015, 13:00 – 14:00

Poster #35.

Title	MGkit: A Metagenomic Framework For The Study Of Microbial Communities
Author	*Francesco Rubino*, Chris Creevey
Affiliation	http://www.aber.ac.uk/en/ibers/
Contact	frr11@aber.ac.uk
URL	https://bitbucket.org/setsuna80/mgkit
License	GPL-2.0

Metagenomics is a relatively new field, in which environmental samples are studied, offering insights into a microbial community as a whole. The wide range of sample types and possible experiments, as well as the scale of the sequencing data, make the creation of new pipelines or the adaptation of a pre-existing one a complex and time consuming task.

Moreover, while metagenomics has been used extensively to study microbial communities from a taxonomic and functional perspective, little has been done to address how the species in a microbiome are adapted to and maintain specific roles in dynamic environments like the rumen. Identifying and assessing the level of this biological adaptation for function is an important aspect that has not been addressed by any currently available metagenomic pipelines.

To address these problems, we have developed a framework that can be used to create and adapt metagenomic analysis workflows, making it faster to implement or prototype different analysis approaches.

Example workflows are included that can scale in size and can be customised with ease. Moreover, we implemented approaches to estimate SNP diversity in metagenomic samples and carry out statistical tests to identify where differences exist, making it possible to apply evolutionary approaches to metagenomic datasets.

The framework does not tie the user to any specific tool, providing templates and documentation that can be used to customise any metagenomic workflow. It is implemented in Python and can be installed on any operating system that supports its library dependencies. The framework is open source, and licensed under GPL-2.0.

BOSC 2015 – *Poster-only Abstracts*
Both Days, 10 & 11 July 2015, 13:00 – 14:00

Poster #36.

From scaffold to submission in a day: a new software pipeline for rapid genome annotation and analysis

Sascha Steinbiss[1], Fatima Silva[2], Brian Brunk[3], Bernardo Foth[1], Christiane Hertz-Fowler[2], Matt Berriman[1], Thomas Dan Otto[1]

[1] Wellcome Trust Sanger Institute, Hinxton, UK. Email: ss34@sanger.ac.uk

[2] University of Liverpool, Liverpool, UK.

[3] University of Pennsylvania, Philadelphia, PA, USA.

Project Website: https://github.com/satta/annot-nf
Source Code: https://github.com/satta/annot-nf
License: ISC (BSD-like)

Technological improvements have enabled genome sequencing and assembly to become efficient and accurate, but this is driving an increased need to annotate newly assembled genomes with the structure and function of genes. These annotations underpin subsequent comparative analyses to identify differences between individual species or strains, such as loss or gain of common and/or species-specific genes and functions.

While established off-the-shelf software solutions for complete genome annotation are readily available for prokaryotes, the need for an efficient eukaryote equivalent remains. Existing heavyweight eukaryotic annotation pipelines are optimized for delivering accurate protein coding gene models but usually do not address partial or non-coding genes, pseudogenes or functional annotations, nor do they generate a product that is ready to submit to public databases (a requirement for publication). The latter involves the preparation of complete annotation results (full gene sets, genomic sequences, protein sequences, functional annotation) in validated and standardized annotation formats (e.g. GFF3, EMBL, GAF). This often manual preparation can result in a substantial bottleneck influencing the total turnaround time to database submission.

We present a new full-stack software pipeline for eukaryotic genome annotation. It accepts input in various states of assembly, covering all stages from pseudochromosome contiguation and gene finding to function assignment. While built on reliable *de facto* standard components such as AUGUSTUS, SNAP (gene finding), RATT (annotation transfer), OrthoMCL (clustering) and GenomeTools (annotation handling), the pipeline includes new and improved versions of existing software such as ABACAS2 (pseudochromosome assembly) as well as bespoke software, e.g. for pseudogene identification. Special care has been taken to make the pipeline produce validated and accurate output even for highly fragmented sequence inputs, as they are common in draft genomes.

The pipeline makes extensive use of modern, state-of-the-art workflow (Nextflow) and deployment technologies (Docker) to ensure scalability, reproducibility and portability for use on powerful stand-alone PCs as well as large compute clusters (e.g. SGE, LSF, SLURM) or cloud platforms (e.g. ClusterK, DNAnexus) with a minimum of effort. In addition, we have created a web-based annotation interface to the pipeline, allowing researchers from the parasitology community to run annotation jobs and comparative analysis tasks on user-provided genomes as well as obtain and visualize the results.

We exemplify the use of the pipeline to annotate a series of new kinetoplastid parasite genomes as well as improve existing parasite annotations.