

Community-driven development for computational biology at Sprints, Hackathons and Codefests

Steffen Möller^{1,2}, Enis Afgan^{3,4}, Michael Banck², Raoul J P Bonnal⁵, Timothy Booth⁶, John Chilton⁷, Peter J A Cock⁸, Markus Gumbel⁹, Nomi Harris¹⁰, Richard Holland¹¹, Matúš Kalaš¹², László Kaján^{2,13}, David R. Powell^{4,14}, Pjotr Prins¹⁵, Jacqueline Quinn¹⁶, Olivier Sallou^{2,17}, Francesco Strozzi¹⁸, Torsten Seemann^{4,14}, Clare Sloggett⁴, Stian Soiland-Reyes¹⁹, Sascha Steinbiss²⁰, Andreas Tille², Anthony J. Travis²¹, Roman Valls Guimera²², Toshiaki Katayama²³, Brad A. Chapman²⁴

¹University of Lübeck, Germany;

²Debian Project;

³Center for Computing and Informatics, Ruđer Bošković Institute (RBI), Croatia;

⁴Life Sciences Computation Centre, Victorian Life Sciences Computation Initiative, Australia;

⁵National Institute of Molecular Genetics, Milan, Italy

⁶NERC Centre for Ecology and Hydrology, Wallingford, UK

⁷University of Minnesota Supercomputing Institute, USA

⁸The James Hutton Institute, UK;

⁹Mannheim University of Applied Sciences, Germany

¹⁰Lawrence Berkeley National Laboratory, Berkeley, CA, USA

¹¹Eagle Genomics Ltd, Cambridge, UK

¹²Computational Biology Unit and Department of Informatics, University of Bergen, Norway

¹³Department of Bioinformatics and Computational Biology, TU Munich, Germany

¹⁴Victorian Bioinformatics Consortium, Monash University, Australia

¹⁵Human Genetics, University Medical Center Utrecht & Laboratory of Nematology, Wageningen University, The Netherlands

¹⁶Autodesk, Inc., USA

¹⁷University of Rennes, France

¹⁸Parco Tecnologico Padano, Lodi, Italy

¹⁹University of Manchester, UK

²⁰Center for Bioinformatics, University of Hamburg, Germany

²¹University of Aberdeen, Institute of Biological and Environmental Sciences,
Aberdeen, Scotland, UK

²²Department of Biochemistry and Biophysics, Science for Life Laboratory,
Stockholm University, Sweden;

²³Database Center for Life Science, Tokyo, Japan

²⁴Harvard Public School of Health, Boston MA, USA,

Email addresses:

SM: steffen.moeller@uksh.de

EA: afgane@gmail.com

MB: mbanck@debian.org

RJPB: lpuccio.febo@gmail.com

TB: tbooth@ceh.ac.uk

PJAC: p.j.a.cock@googlemail.com

MG: m.gumbel@hs-mannheim.de

NH: nlharris@gmail.com

RH: dicknetherlands@gmail.com

MK: matus.kalas@uib.no

LK: lkajan@rostlab.org

DRP: david.powell@monash.edu

PP: pjotr2014@thebird.nl

JQ: jyquinn@gmail.com

OS: osallou@debian.org

FS: francesco.Strozzi@tecnoparco.org

TS: torsten.seemann@monash.edu

CS: claresloggett@gmail.com

SSR: soiland-reyes@cs.manchester.ac.uk

SS: steinbiss@gmail.com

AT: tille@debian.org

AJT: tony.travis@abdn.ac.uk

RVG: brainstorm@nopcode.org

TK: ktym@dbcls.jp

BAC: chapmanb@50mail.com

Abstract

Background

Computational biology comprises a wide range of different technologies and approaches. In order to combine different technologies to create more powerful workflows, multiple individuals and research groups generally need to collaborate. The format of data exchanged, the biology behind that data, and the individuals contributing the data or providing tools for its interpretation need to find mutual understanding and consensus. Much conversation and joint investigation are required in order to identify and implement the best approach to using and combining the latest technologies.

Traditionally, scientific conferences feature talks that present novel technologies or insights, followed up by informal discussions during coffee breaks - not necessarily involving the presenting party. In multi-institution collaborations, in order to reach agreement on implementation details or to transfer deeper insights in a technology and practical skills, a representative of one collaborating research group typically visits the other. However, this does not scale well when the number of technologies or research groups is large.

Conferences have responded to this issue by introducing Birds-of-a-Feather (BoF) sessions, which offer an opportunity for individuals sharing interests, commonly with prior acquaintance on mailing lists, to intensify their interaction. However, parallel BoF sessions often make it hard for participants to join multiple BoFs and find common ground between the different technologies. Also, industrial participants in these discussions are rare, since their focus is on selling a product, not in forming collaborations that may not be immediately profitable.

Results

We here summarise our experience with Codefests, Hackathons and Sprints, which are interactive developer meetings in our field that aim to reduce the limitations of traditional scientific meetings described above, by strengthening the interaction among peers and letting the participants determine the schedule and topics covered at the meeting. These meetings are commonly run as loosely scheduled “unconferences” [\(self-organized identification of participants and topics for meetings\)](#) over at least two days, with early introductory talks to welcome and schedule contributors, followed by intensive collaborative coding sessions. We describe some differences in the way these meetings are organised, the audience is addressed, and in their outreach to their respective communities. The most prominent achievements of those meetings are summarised.

Conclusions

Hackathons, Codefests and Sprints share a stimulating atmosphere that encourages participants to jointly brainstorm and tackle problems of shared interest in a self-driven proactive environment.

Background

Sprints, Hackathons and Codefests are all names for informal software developer meetings, especially popular in Open Source communities. These meetings, which often take place in loose conjunction with more traditional conferences, are a vital part of the international network of interactions between software developers working in bioinformatics and computational biology, and complement purely online interactions such as project mailing lists, online chat, web forums, voice and video calls.

Open Source software has figured significantly in bioinformatics for over 20 years. Leveraging and building upon existing Open Source software is a powerful way to

rapidly implement new ideas and methods into reliable working code. This helps in a world where scientific groups are under increasing pressure to produce results quickly and more cheaply than ever. The challenge for everyone is to be aware of existing implementations of a particular desired functionality and their compatibility with local infrastructure. Strategically, it is beneficial to know other contributors to externally maintained libraries, and to ensure that contributions are integrated with the remaining code in the best future-compatible way and with the least possible redundancies. This paper summarises the activities and backgrounds of three related types of meetings: Sprints, Hackathons and Codefests. These share the aim of fostering collaborative interactions and the trust to allow mutual dependencies between developers in computational biology and bioinformatics. Although these meetings share common features, each event has its own particular slant and flavour of the community.

Methods

BioHackathons

A series of BioHackathons (short for “biologically motivated code hacking marathons”) have been held. The BioHackathons (Katayama et al. 2010, 2011, 2013) have been organized as an invitational event with the loose intention of encouraging the participants to collaborate on a given theme. This flexibility recognises that with hindsight the most productive results/ideas were not always predictable beforehand, but emerged from self-organized collaborative work during the BioHackathons when developers from different domains spent a week talking and coding together. The original BioHackathons in 2002 and 2003 were mainly dedicated to interoperability in handling sequence data amongst the Bio* projects. BioPerl, BioJava, BioPython, and BioRuby groups worked together to develop common

sequence object models, APIs for the BioSQL database and Web services. This ensured that fundamental bioinformatic functionality would be compatible among those four programming toolkits. The first BioHackathon resulted in the BioPerl publication (Stajich JE et al., 2002).

The first few BioHackathon meetings in Japan focused on Web services and interoperability (Katayama T et al., 2010 and 2011) and later moved to improving life science data integration with Semantic Web technologies (Katayama T et al., 2013), reflecting the perceived needs of the biomedical community to move from workflows towards integration of data resources, ontology, semantics and reasoning.

BOSC Codefests

The Bioinformatics Open Source Conference (BOSC) was established in 2000 by the Open Bioinformatics Foundation Bio* project members as an international venue for showcasing new projects and progress, and for developers worldwide to meet in person. Since then BOSC has been held yearly as a special interest group (SIG) meeting preceding the annual Intelligent Systems in Molecular Biology (ISMB) conference, one of the most popular bioinformatics conferences. Since 2010 the annual BOSC meetings have also included a two-day informal Codefest, which has typically attracted between 20 and 35 individuals. Meeting in person is a valuable complement to traditional online distributed teamwork, allowing more intensive discussions and social bonding that continues into the BOSC meeting.

Over 30 developers participated in the BOSC 2013 Codefest, hosted by Humboldt-Universität zu Berlin. Projects accomplished by attendees included the extension of several existing open-source tools, development of standards for provenance tracking, and integration of infrastructure management, visualization and parallelization frameworks (Chapman 2013). A key outcome was increased interoperability between

tools, an essential requirement for carrying out large scale science in rapidly evolving research areas (Harris et al., 2013).

The BOSC 2013 Codefest accomplishments included small updates in the Biopython and Cloud Bio-Linux projects, and work on practical issues like an integration of SLURM, an HPC job manager, next to the IPython Cluster. As for all projects, and for all such events, there was a review of documentation and introduction of new users to the community.

Anyone with a particular attraction to a language of another programming philosophy, e.g., for functional programming, would require a considerable effort to reach the same functionality as a general workhorse. The 2013 Codefest introduced ScaBio, which was integrated into BioJava. It is written in Scala, another language that runs on top of the Java Virtual Machine. The two languages are mutually compatible, which ensures that no redundancy between the two projects is required. Projects can mix the two languages, choosing the one most suited to the particular problem at hand.

The BioRuby group tackled a new project which enables programmers to develop Web applications for BaseSpace (<http://basespace.illumina.com>). BaseSpace is a cloud solution provided by Illumina, a company that makes DNA sequencers on which users can apply various analysis tools to their next generation sequencing (NGS) data. During the 2013 Codefest, a Ruby version of the BaseSpace SDK was tested, documented and completed for release as an Open Source package. This project will bring benefits for the Open Source community in academia and industry. BioRuby/Biogem developers can provide their work also as BaseSpace apps to increase awareness and impact on their platform. Illumina in turn raises the value of BaseSpace. Most importantly, users of Illumina instruments can apply a variety of

methods easily without learning any programming language. Recently, the BaseSpace Ruby SDK was contributed to Illumina as one of the official toolkits along with the Python, Java, C# and R versions with the help of a Code participant from Illumina. Mobyly (Neron B et al., 2009) was modified to provide provenance data using W3C PROV-O (Lebo T et al., 2013) and integrated the EDAM ontology (Ison J et al., 2013) for describing its services. The Synthetic Biology Open Language Visual Standard (Quinn J et al., 2013) had been formalized as an ontology, and the 2013 Codefest brought together Semantic Web engineers who were consulted on how to model the visual representation of terms within the ontology.

Sometimes at Codefests existing technologies are retired in favor of new ones. For the last 8 years, RNAmmer has been the standard tool for predicting ribosomal RNA features in genomes. Its drawbacks are that it relies on small, older databases; requires an older conflicting version of HMMER; and has restrictive licence terms. To resolve these issues for prokaryotes, a new rRNA predictor was implemented which uses the new “nmmer” tool from HMMER 3.1 for searching DNA profiles against DNA sequences. This led to the development of Barrnap (<http://www.vicbioinformatics.com/software.barrnap.shtml>), a small Perl script that takes FASTA as input, and outputs the rRNA feature predictions in GFF3 format. The two will be packaged in Bio-Linux and will replace RNAmmer in the Prokka bacterial annotation system (Seemann T, 2014). The identification of the problem, a technical solution and its immediate employment in larger workflows need several individuals to work together to agree on the approach and begin the required coding--a type of interaction fostered by Codefests and similar events.

Sprints

When Linux surfaced as a free operating system, it was adopted quickly by the research community, including many bioinformatics developers. Consequently much of the software developed for Linux was released with Free/Open licenses, much like the public data policy adopted by major international biological databases (Fernández-Suárez XM and Galperin MY, 2013). For example, the Open Source Bio* projects, most prominently BioPerl (Stajich JE et al., 2002), BioJava (Prlić A et al., 2012), Biopython (Cock PJ et al., 2009) and BioRuby (Goto N et al., 2010), originated over 14 years ago as community projects providing widely used libraries for building bioinformatics tools, pipelines and one-off analysis scripts.

To help promote the redistribution, general availability and mutual compatibility of software, the Debian Linux project launched the Debconf event, a hacking session right before the Debian Conference. Debconf is an unconference, a meeting at which people meet and work on specific topics, either alone or in teams. Along the same lines, the Debian Med initiative was founded in 2001 (Tille A, 2001), with the Sprint as an annual meeting.

Debian Med (Möller S et al., 2010) and Bio-Linux (Field D et al., 2006) provide the necessary infrastructure for distributing software tools and their updates to the wider community. This is achieved by packaging and distributing the tools in the context of these larger tool repositories. The Cloud Bio-Linux community in turn further enhances the distribution, tailored for bioinformatics on cloud infrastructure (Krampis K et al., 2012). The Sprint helps with the integration of these efforts.

The 2013 Debian Sprint invited contributors to BOINC (Anderson DP, 2004), a distributed computing project, and we found the binaries of Debian, auto-compiled for multiple different architectures, to be directly usable in BOINC projects across Linux distributions. Together with an ongoing effort to provide packages for the BOINC

server side, this will help increase the availability of compute time for biological groups, e.g., for biochemists with a novel receptor addressing protein docking (Balan DM et al., 2012). This is a good example of the positive outcomes that can result from the cross-disciplinary collaborative activities at Sprints and similar events.

The integration of a single software package can trigger a collection of multiple other software tools to be packaged together to complete common workflows in that field. For example, porting of the GenomeTools software suite (Gremme G et al., 2013) during the 2012 Debian Med Sprint not only integrated a substantial number of published and established sequence analysis tools (see <http://genometools.org> for a complete list) into Debian, but also paved the way for future inclusion of packages dependent on the associated GenomeTools library, e.g., ParsEval (Standage D et al., 2012) or LTRsift (Steinbiss S et al., 2012). This emphasizes the role of Sprints as strong promoters of synergistic effects within the community.

Results

Hackathons and similar types of developer meetings can have a major impact on the community. Having 20+ talented and motivated individuals with shared interests work together for two or more days can be extremely productive. Each person who participates in these collaborative meetings brings their own complement of technical, scientific and social strengths. These developer meetings allow for cross-pollination of skills between separated silos of expertise. These events can be organised around any topic with a large enough user base. They combine individualised training, social networking and technical contributions and help pave the way toward new scientific discoveries. Results include real software solutions, documentation, and the joint communication of milestones for future developments.

Each event described in this paper has its own culture and organization. The Japanese BioHackathon events are the longest, at one week each, and this series is also the longest running and best-funded, with support for travel of international participants. It has already yielded several journal publications. The Debian Med Sprints are noteworthy for bringing upstream developers (who are directly working on scientific Free Software projects) and downstream scientific users and developers (who are working on Debian Med derived distributions) together, along with core and prospective Debian Med members. This heterogeneous group allowed for more efficient skill exchange and a wider array of topics. Everybody can weigh in with their particular strength, while letting the rest of the group focus on their particular interests and strong points.

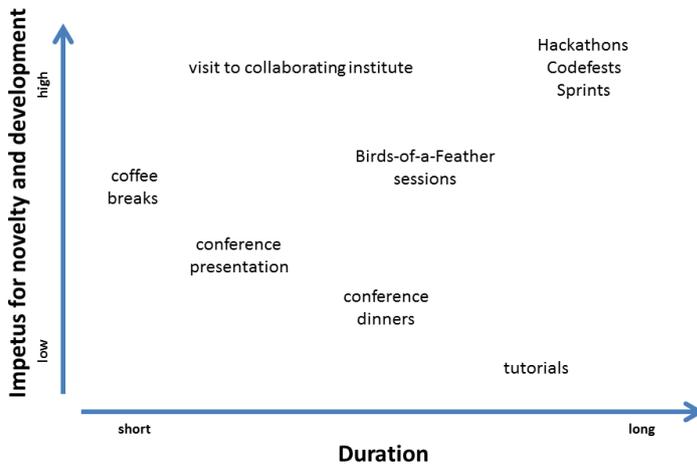


Figure 1 - Forms of academic exchange

The most common opportunities for scientists to meet separated by their drive for

novelty and development (Y axis) and their duration (X axis), the prior as an undisputed but very subjective consensus among the authors.-Sprints, Codefests and Hackathons are dominating for their focus on joint new developments, the transfer of

expertise for new scientific questions, the distribution of infrastructure and a network of trust between the contributors. *A Summer of Code combines many of the other concepts over a long time and may treat those who learn different from those who share their insights, which renders direct comparisons difficult.*

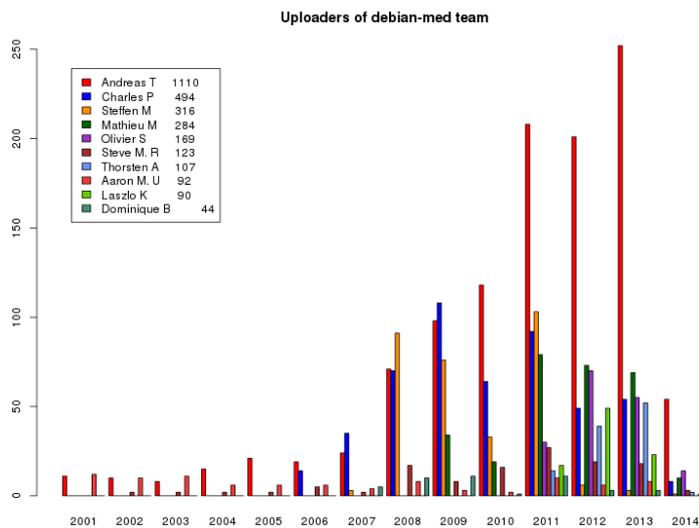


Figure 2: Number of Uploads to Debian Med per individual

The figure from <http://blends.debian.net/liststats/> indicates the activity of team members with upload privileges. One clearly sees the increased breadth since the first Sprint in early 2011.

Formatiert: Schriftart: Fett

Formatiert: Schriftart: Kursiv

Formatiert: Schriftart: Kursiv

Formatiert: Schriftart: Kursiv

Discussion

Since Open Source software developers spread across the globe already collaborate by communicating online via distributed source-code repositories, mailing lists, chat and other means, the time and expense of travelling to meet up in person may seem like a waste. However, physical meetings bring an edge to productivity, including temporarily avoiding day to day workplace duties, and the opportunity to see software

and infrastructure problems from outside your local needs. Also, meeting in person temporarily solves the problems of cross time zone collaborations. This is particularly important for contributors in Australasia communicating with Europeans or Americans, where live interactions like conference calls must be often scheduled outside normal office hours, and any conversation by email can take days. Meeting physically also helps build interpersonal relationships and can motivate attendees to follow up on issues they might not tackle otherwise. Most people feel more of a sense of connection and commitment to people they have worked with in person than to those they have never met, and the networking also has a career advantage (Stobbe et

al., 2013). The concept of a Summer of Code as e.g. run by Google or the European Space Agency, combines remote collaborations with a summits after the projects are completed. Many if not most of the contributors to the here outlined Open Source meetings also mentor for the Google Summers of Code – this is of no competition, but much like extra resources for the researchers’ interest and, if attending the summit, an opportunity to look deeply beyond one’s personal area of expertise.

~~The All~~ here outlined events are focusing on the science and easier access to technology – in an intense and unique way, open to everyone (Figure 1). It was shown (<http://blends.debian.net/liststats/>) for the Sprints that the meetings have a direct effect on the activity of the projects, both in terms of the number of patches submitted and, on the social side, the number of emails distributed on the mailing list.

A considerable change observable with the advent of Synthetic Biology is that individuals with a first degree close to engineering or computer science are building and using tools for tasks that have in the past been performed manually by biologists, e.g. planning of cloning experiments. We need to learn to address and attract such neighboring communities to co-develop and share Open Source infrastructure to avoid

being crowded out by closed-source solutions. A critical mass of software solutions and users for the synthetic biology field has yet to emerge, and the interplay between Open Source and commercial entities is yet to be established. The BOSC 2013 Codefest (Harris NL et al., 2013) helped establish first contact between the Bioinformatics and Synthetic Biology communities, and hopefully will lead to helpful interchange between the fields.

Not only academic or research institutions use Open Source tools or frameworks. Open Source tools are widely used in commercial research and commercial service providers who build products and services around Open Source components. Both these types of companies have an interest in improving the quality of the Open Source software they use. Hackathons and Codefests offer an opportunity for these improvements to be made whilst simultaneously meeting with the original developers, learning from them, and giving them guidance as to future requirements.

The BOSC Codefest and the Sprint are more constrained as a two day meeting than the week-long Hackathon. As a consequence, there is less time for the participants to be trained or self-educated to better sync with a larger group. Here, the Google Summer of Code with its months long individualised training has an advantage. Another concern for long term effects of the meeting on the participants' are the constraints by daily routine. For the time of the event, the Summer of Code achieves such by money and public peer pressure. The here described events mostly select for individuals who already arranged themselves with Open Source development schemes.

The motivation for small companies to get involved with an Open Source Hackathon or Codefest is typically based on the expectation that the participation will have a positive effect on the perception of the companies' products and in anticipation of

additional sales-- whether they are taking part actively or simply sponsoring the event. Large research corporations that use Open Source products internally may want to get involved only if the event will develop features and fix bugs that will improve the company's internal productivity, and hence save operational costs. For a service or product provider that uses Open Source tools, it will only want to participate if the outcome is an improved tool or feature set that they can then build commercial offerings around for their own customers, or use to improve their own internal processes and reduce overheads. To attract greater participation from commercial partners, Hackathons and Codefests must therefore include a certain amount of applied research driven by the requirements of these partners, and be willing to guide their development efforts in a direction that will deliver commercial value. This may jar with participants from a pure academic research background where commercial requirements are a much lower priority, but it is essential to gain external sponsorship.

For computational biology, the underlying infrastructure is mostly Open Source because of the historic freeness of the sequence data. On the applications' side, however, e.g. for the assembly and optimisation of workflows, for which the Assemblathon (Earl D. et al., 2011) may stand representatively, or user centric design (Pavelin K. et al., 2012), the license of widely distributed tools is not of concern. Diversity of participants may be distracting or fruitful, e.g. for the OpenDataDay [<http://opendataday.org>] – one just needs a joint theme integrating everyone - and that we have.

Conclusions

Participation in a Hackathon, Codefest, or Sprint can be an extremely rewarding experience for the participants and the greater community. These informal, interactive meetings have played an important role in the development of Open Source

technologies and the community. At these events, developers of all ages and levels of experience interact with each other. Besides the joint problem-solving work, this has consequences on how the projects present themselves to encourage new contributors to surface. Hands-on training and exchange of experiences, actively and passively, remain core features of the events.

In closing, we can point to specific examples of software developments and bug fixes made during the developer meetings described, and in some cases publications that have resulted from these meetings. However, their true worth is more intangible in the form of the community itself, new and strengthened collaborations, and the spread of ideas and best practices--both scientific and for software development.

Authors' contributions

All authors contributed to the preparation of the document and participated in at least one of the here described events.

Acknowledgements

All scientific groups are thanked that support contribution to Open Source software.

The events' hosting institutions and funding agencies are thanked for providing facilities and support. O'Reilly, Electric Genetics, Apple Asia, the Debian project and the employing institutions are thanked for contributing to the travel costs of the events. The authors are listed alphabetically by surname, with the exception of the first and last authors who initiated the meetings. [The anonymous reviewers are thanked for their constructive comments.](#)

References

1. Anderson DP (2004) **BOINC: A System for Public-Resource Computing and Storage**. Proceedings of the 5th IEEE/ACM Workshop on Grid Computing, pp. 4-10. doi:10.1109/GRID.2004.14
2. Balan DM, Malinauskas T, et al. (2012) **High-throughput Molecular Docking Now in Reach for a Wider Biochemical Community**. *PDP 2012*: 617-621
3. Bonnal RJ, Aerts J, et al. (2012) **Biogem: an effective tool-based approach for scaling up open source software development in bioinformatics**. *Bioinformatics*. **28** (7):1035-1037. doi: 10.1093/bioinformatics/bts080
4. Cock PJ, Antao T, et al. (2009) **Biopython: freely available Python tools for computational molecular biology and bioinformatics**. *Bioinformatics*. **25** (11):1422-1423. doi: 10.1093/bioinformatics/btp163
5. Earl D, Bradnam K, et al. (2011) **Assemblathon 1: a competitive assessment of de novo short read assembly methods**. *Genome Res*. **21**(12):2224-2241. doi: 10.1101/gr.126599.111
65. Fernández-Suárez XM and Galperin MY (2013) **The 2013 Nucleic Acids Research Database Issue and the online molecular biology database collection**. *Nucleic Acids Res*. **41** (Database issue):D1-7. doi: 10.1093/nar/gks1297
76. Field D, Tiwari B, et al. (2006) **Milo: Open software for biologists: from famine to feast**, *Nat Biotechnol*. **24**:801-803. doi:10.1038/nbt0706-801
87. Filangi O, Beausse Y, et al. (2008) **BioMAJ: a flexible framework for databanks synchronization and processing**. *Bioinformatics*. **24** (16):1823-1825. doi: 10.1093/bioinformatics/btn325.

Formatiert: Schriftart: Fett

Formatiert: Schriftart: Kursiv

- ~~98~~. Goto N, Prins P, et al. (2010) **BioRuby: bioinformatics software for the Ruby programming language.** *Bioinformatics*. **26** (20):2617-2619. doi: 10.1093/bioinformatics/btq475
- ~~910~~. Gremme G, Steinbiss S, et al. (2013) **GenomeTools: a comprehensive software library for efficient processing of structured genome annotations.** *IEEE/ACM Trans. Comp. Biol. Bioinf.* **10** (3):645-656. doi: 10.1109/TCBB.2013.68
- ~~119~~. Harris NL, Cock PJA, Chapman BA, Goecks J, Hotz H et al. (2013) **The Bioinformatics Open Source Conference (BOSC) 2013.** PeerJ PrePrints 1:e83v1. doi:10.7287/peerj.preprints.83v1
- ~~124~~. Ison J, Kalaš M, et al. (2013) **EDAM: an ontology of bioinformatics operations, types of data and identifiers, topics and formats.** *Bioinformatics*. **29** (10):1325-1332. doi: 10.1093/bioinformatics/btt113
- ~~132~~. Kaján L, Yachdav G, et al., (2013) **Cloud Prediction of Protein Structure and Function with PredictProtein for Debian.** *BioMed Research International*. **2013**:398968. doi:10.1155/2013/398968
- ~~143~~. Katayama T, Arakawa K, et al. (2010) **The DBCLS BioHackathon: standardization and interoperability for bioinformatics web services and workflows.** The DBCLS BioHackathon Consortium*. *J Biomed Semantics*. **1** (1):8. doi: 10.1186/2041-1480-1-8.
- ~~154~~. Katayama T, Wilkinson MD, et al. (2011) **The 2nd DBCLS BioHackathon: interoperable bioinformatics Web services for integrated applications.** *J Biomed Semantics*. **2** (2):4. doi: 10.1186/2041-1480-2-4.

165. Katayama T, Wilkinson MD, et al., (2013) **The 3rd DBCLS BioHackathon: improving life science data integration with Semantic Web technologies.** *J Biomed Semantics.* **4** (1):6. doi: 10.1186/2041-1480-4-6.
176. Krampis K, Booth T, et al. (2012) **Cloud BioLinux: pre-configured and on-demand bioinformatics computing for the genomics community.** *BMC Bioinformatics.* **13**:42. doi: 10.1186/1471-2105-13-42.
187. Lebo T, Sahoo S, McGuinness D, Belhajjame K, Cheney J, Corsar D, Garijo D, Soiland-Reyes S, Zednik S, Zhao J. (2013): **PROV-O: The PROV Ontology; W3C Recommendation** 30 April 2013
[<http://www.w3.org/TR/2013/REC-prov-o-20130430/>]
198. Möller S, Krabbenhöft HN, et al. (2010) **Community-driven computational biology with Debian Linux.** *BMC Bioinformatics.* **11** (S-12): S5.
- ~~2019.~~ Quinn J, Beal J, ~~Bhatia S, Cai P, Chen J, Clancy K, Hillson N,~~ et al. (2013): "Synthetic Biology Open Language Visual (SBOL Visual), version 1.0.0." *BioBricks Foundation RFC 93.* doi:1721.1/78249
- ~~21.~~ Katrina Pavelin, Jennifer A. Cham (2012) Bioinformatics Meets User-Centred Design: A Perspective. *PLoS Comput Biol.* 8(7):e1002554 doi: 10.1371/journal.pcbi.1002554
- ~~220.~~ Prlić A, Yates A, et al. (2012) **BioJava: an open-source framework for bioinformatics in 2012.** *Bioinformatics.* **28** (20):2693-2695. doi: 10.1093/bioinformatics/bts494
- ~~2123.~~ Seemann T (2014) **Prokka: rapid prokaryotic genome annotation.** *Bioinformatics, in press.*

Formatiert: Französisch (Frankreich)

Formatiert: Block

Formatiert: Englisch (USA)

Formatiert: Englisch (USA)

Formatiert: Schriftart: Kursiv

Formatiert: Englisch (USA)

Formatiert: Englisch (USA)

- ~~22~~24. Stajich JE, Block D, et al. (2002) **The Bioperl toolkit: Perl modules for the life sciences**. *Genome Res.* **12** (10):1611-1618. doi: 10.1101/gr.361602
- ~~23~~25. Standage DS and Brendel VP (2012). **ParsEval: parallel comparison and analysis of gene structure annotations**. *BMC Bioinformatics* **13**:187.
doi:10.1186/1471-2105-13-187
- ~~24~~26. Steinbiss S, Kastens S, et al. (2012) **LTRsift: a graphical user interface for semi-automatic classification and postprocessing of de novo detected LTR retrotransposons**. *Mobile DNA* **3**:18. doi:10.1186/1759-8753-3-18
- ~~25~~27. Stobbe M, Mishra T, Macintyre G (2013) **Breaking the Ice and Forging Links: The Importance of Socializing in Research**. *PLoS Comput Biol* **9**(11): e1003355.
- ~~28~~6. Tille A (2001) **Packaging Software for Use in Medicine**. Libre Software Meeting, Bordeaux. France

Formatiert: Einzug: Links: 0 cm,
Erste Zeile: 0 cm