Bioconductor Annual Report

Martin Morgan
Fred Hutchinson Cancer Research Center

22 July, 2015

Contents

1 Project Scope

Bioconductor provides access to software for the analysis and comprehension of high throughput genomic data. Packages are written in the *R* programming language by members of the *Bioconductor* team and the international community. *Bioconductor* was started in Fall, 2001 by Dr. Robert Gentleman and others, and now consists of >1024 packages for the analysis of data ranging from sequencing to flow cytometry.

Table 1: *Bioconductor*-related funding

	Award	Start	End
Active			
NHGRI / NIH	2U41HG004059	9/26/2011	2/29/2016
NCI / NIH	1U24CA180996	9/1/2014	8/31/2019
NSF	1247813	8/1/2013	7/31/2015
Participating			
EC-H2020	SOUND	9/1/2015	8/31/2018
Pending			
NHGRI / NIH	2U41HG004059	3/1/2016	2/28/2021

1.1 Funding

Funding is summarized in Table 1.

The project is primarily funded through National Human Genome Research Institute award 2U41HG004059 (Community Resource Project; Morgan PI, with Carey and Irizzary), 'Bioconductor: An Open Computing Resource for Genomics'. Current funding expires 2/29/2016; a renewal has been through scientific review (July 7, 2015) with an initial impact score of 10 (possible scores range from 10 (good) to 90 (less good)). Summary statements are not yet available. The program officer has not yet been in contact.

The project receives additional funding through 1U24CA180996 (Morgan PI, with Carey, Hansen, Waldron), 'Cancer Genomics: Integrative and Scalable Solutions in *R* / *Bioconductor*'.

Funding from NSF 1247813 (BIGDATA; Morgan, PI, with Carey, Huber, Taylor [Galaxy]), 'BIGDATA: Mid-Scale: DA: ESCE: Collaborative Research: Scalable Statistical Computing for Emerging Omics Data Streams' expired 7/31/2015.

European Commission Horizon 2020 project 633974 (Huber, PI, with Morgan and others), 'SOUND: Statistical multi-Omics UNDerstanding of Patient Samples' has significant *R* / *Bioconductor* components.

Funding supports 7-8 full-time personnel at FHCRC, plus additional individuals at subcontract sites; see section 3.3.

1.2 Package and Annotation Resources

R software packages represent the primary product of the *Bioconductor* project. Packages are produced by the *Bioconductor* team and from international contributors. Table 2 summarizes growth in the number of packages hosted by *Bioconductor*, with 1024 software packages available in release 3.1. The project produces 883 'annotation' packages to help researchers place analytic results into biological context. Annotation packages are curated resources derived from external data sources, and are updated at each release. The project has developed, over the last year, the 'AnnotationHub' resource for serving and managing genome-scale annotation data, e.g., from the Roadmap Epigenomics project, NCBI, and Ensembl. There are 34881 records in the current hub.

Software packages were downloaded by 286,305 unique non-FHCRC IP addresses between August, 2014 and July, 2015, approximately 22% more than in the previous year.

1.3 Courses and Conferences

Course and conference material and announcements for upcoming events are available. Courses and conferences with significant input from key *Bioconductor* personnel have been held in the following worldwide locations in the last year:

Table 2: Number of contributed packages included in each *Bioconductor* release. Releases occur twice per year.

Release		N	Release		N	Release		N	Release		N
2002	1.0	15	2006	1.8	172	2010	2.6	389	2014	2.14	824
	1.1	20		1.9	188		2.7	419		3.0	936
2003	1.2	30	2007	2.0	214	2011	2.8	467	2015	3.1	1024
	1.3	49		2.1	233		2.9	517			
2004	1.4	81	2008	2.2	260	2012	2.10	554			
	1.5	100		2.3	294		2.11	610			
2005	1.6	123	2009	2.4	320	2013	2.12	671			
	1.7	141		2.5	352		2.13	749			

Table 3: Support site visitors from October, 2014. Users: registered users visiting during the reporting period; Visitors: Google analytics visitors during the reporting period. 2014-15 spans 10-months.

Year	Users	Visitors	Posts	Replies
2014-15	2179	122,332	2169	6535

- *BioC 2015* – July, Fred Hutchinson Cancer Research Institute, Seattle, WA, USA.
- *Introduction to Bioconductor for Sequence Analysis – useR!* June, Aalbourg, Denmark
- *Computational Statistics for Genome Biology (CSAMA)* – CSAMA, June, Brixen-Bressanone, Italy
- *Use R / Bioconductor for Sequence Analysis (Intermediate Course)* – April, Seattle, WA, USA.
- *MOOC: Statistics and R for the Life Sciences* (Irizzary, Carey)
- *BioC Europe 2015*, Heidelberg, Germany.
- *Learning R / Bioconductor for Sequence Analysis* – October, Seattle, WA.
- *EMBO Practical Course on Analysis of High-Throughput Sequencing Data* – October, Hinxton, UK.
- *Brazilian Cancer Epigenomics Workshop* – August, Ribeirão Preto, Brazil.

1.4 Community Support

The project transitioned from a user mailing list to support site in October, 2014. There are about 185 new 'top-level' posts and 595 comments or answers per month. The number of (google analytics) weekly sessions have grown from about 3000 per week at introduction to about 11000 per week in July, 2015. Statistics are summarized in Table 3. Mailing list statistics are provided in Table 4.

We continue to provide bioc-devel, a mailing list forum for package contributors' questions and discussion relating to the development of *Bioconductor* packages. There are 1013 subscribers on this list.

All lists provide a means of disseminating project news and a space for members of the community to share their knowledge about use of *Bioconductor* packages and best practices for data analysis. Table 4 lists the number of posts and number of unique authors as a monthly average since 2002.

Web site access is summarized in Figure 1. The web site served 1.348M sessions (508,930 unique visitors) from July 1, 2014, through June 30, 2015 (statistics from Google Analytics). Visitors come from the United States (35%), China (7.9%), the United Kingdom (7.2%), Germany (6.7%), France (3.2%), Canada, India, Japan, Spain, Italy, and 210 other countries. Unique visitors grew by 26%.

Table 4: Monthly average number of posts and number of unique authors for the `bioconductor` mail list from January, 2002 – January, 2014.

Year	Posts per month	Authors per month	Year	Posts per month	Authors per month
2002	59	13	2009	450	86
2003	231	47	2010	504	170
2004	320	60	2011	467	166
2005	353	61	2012	597	195
2006	348	59	2013	569	204
2007	432	75	2014	498	179
2008	424	83			

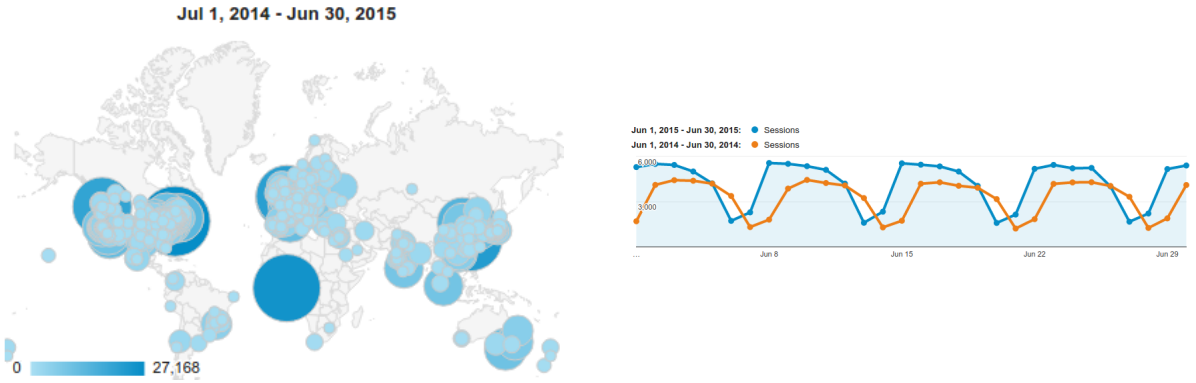

Figure 1: *Bioconductor* Access Statistics, 2014. Left: international visits. Right: Web site access, June 2013 (orange) and 2014 (blue).

1.5 Publication

Bioconductor has become a vital software platform for the worldwide genomic research community. Table 5 summarizes PubMed author / title / abstract or PubMedCentral full-text citations for 'Bioconductor'.

Featured and recent publications citing *Bioconductor* are available on the *Bioconductor* web site, and are updated daily. Important recent publications from the overall project include Huber et al., 2015, *Orchestrating high-throughput genomic analysis with Bioconductor*, Nature Methods 12:115-121 and Lawrence et al., 2013, *Software for Computing and Annotating Genomic Ranges*, PLoS Comput Biol 9(8): e1003118.

Table 5: PubMed title and abstract or (2012 and later) PubMedCentral full text searches for "Bioconductor" on publications from January, 2003 – July, 2015.

Year	N	Year	N	Year	N	Year	N
2003	7	2007	44	2011	68	2015*	1253
2004	13	2008	52	2012	1386		
2005	19	2009	62	2013	2048		
2006	30	2010	52	2014	2401		

2 New Accomplishments

2.1 Software

GenomicRanges represents a mature infrastructure for working with sequence data. Implementation of 'nested containment lists' substantially enhances memory use and speed associated with a central operation (finding overlaps between millions of ranges).

Rhtslib provides *Bioconductor* developers with access to state-of-the-art BAM file manipulation, including on Windows. *Rsamtools* will (despite its name) be updated in the very near future to use *Rhtslib*.

AnnotationHub provides access to genome-scale resources, as illustrated in the How-To Vignette.

BiocParallel provide a consistent and increasingly comprehensive parallel computing interface across cores, computers, clusters, and the cloud. *GenomicFiles* provides infrastructure for iterating through and processing in parallel genome-scale files and collections of files.

2.2 Infrastructure

Virtualization Docker and Amazon Machine Instance images are available.

Github hosts a mirror of our SVN repository of all *Bioconductor* packages. We have revised the git-svn bridge faciliites to ease developer contributions through github.

Shields on package landing pages (e.g., *Rsamtools*) provide users and developers with additional insight and inspiration on package use and robustness, including unit test and coverage metrics.

2.3 User Support

Support site replaces our user mailing list.

Course Materials organize and make much more accessible recent course and training material.

Quarterly Newsletters provide users and developers with insight into project developments.

MOOCs offered by Irizzary, Carey, and colleagues have reached 10,000's of people.

biocViews have been more heavily curated to enhance utility in package discovery.

Videos have been explored as a training mechanism.

Workflows provide cross-package training material; Huber has extended this concept with the recently launched F1000 *Bioconductor* channel.

3 Core Tasks & Capabilities

3.1 Core Tasks

1. Package Building and Testing. The *Bioconductor* project provides access to its packages through repositories hosted at `bioconductor.org`. One of the services provided to the *Bioconductor* community is the automated building and testing of all packages. Maintaining the automated build and test suite and keeping the published package repositories updated requires a significant amount of time on the part of the Seattle *Bioconductor* team. As the project has grown, the organizational and computational resources required to sustain the package build system have also increased; see section 3.2.
2. Package Dissemination.
3. Software Development.
4. End-User and Developer Support.
5. New Package Submission. The *Bioconductor* project relies on technical review process of candidate packages to ensure they contain high-quality software. The Seattle *Bioconductor* team spends a considerable amount of time managing new contributions by previewing the software for quality, managing peers during the review

process to ensure scientific relevance, and communicating with the software developers on what steps need to be taken for their contribution to be included within *Bioconductor*. From August, 2014 – July, 2015, approximately 291 software packages have been managed by the Seattle *Bioconductor* team.

6. Annotation Data Packages. The *Bioconductor* project synthesizes genomic and proteomic information available in public data repositories in order to annotate genomic sequences and probes of standard microarray chips. These annotation data packages are made available to the community and allow *Bioconductor* users to easily access meta data relating to their experimental platform. We maintain automated tools to parse the available information. Due to quickly changing data standards, the maintenance of the code used to produce the annotation packages requires constant attention. Work during the recent release cycles has focused on flexible approaches to transitioning from gene-level annotations relevant for expression arrays to genome coordinate annotations that form the basis of sequence-based annotations.
7. Semi-Annual Releases.

3.2 Hardware and Infrastructure

The *Bioconductor* project provides packages for computing platforms common in the bioinformatics community. We provide source packages that can be installed on Linux and most UNIX-like variants, as well as binary packages for Windows and OS X. To ensure that packages are consistently documented, easy to install, and functioning properly, we run a nightly build during which we test all packages in the release and development repositories.

The build system currently consists of at least two Windows machines, two Linux machines, and two MacOS machines. The web site, support site, AnnotationHub, and additional servers are hosted on virtual machines, some of which are Amazon machine instances. The build machines are heavily taxed, and the overall architecture of our build system (complete nightly builds) leave little room for growth.

3.3 Key Personnel

The **Core Development Team** are employees of the Fred Hutchinson Cancer Research Center, developing software and other infrastructure and ensuring day-to-day operation of the project. Core team members in the period covered by this report incude Martin Morgan, Sonali Arora, Marc Carlson, Nathaniel Hayden, James Hester, Hervé Pagès, Valerie Obenchain, Dan Tenenbaum, and Paul Shannon, developer (20% time).

The **Technical Advisory Board** provides guidance through monthly telephone conference calls. Current members include: Vincent Carey, Brigham & Women's; Aedin Culhane, Dana-Farber Cancer Institute; Sean Davis, National Cancer Institute; obert Gentleman, 23andMe; Kasper Daniel Hansen, Bloomberg School of Public Health, Johns Hopkins University; Wolfgang Huber, European Molecular Biology Laboratory, Heidelberg, Germany; Rafael Irizarry, Dana-Farber Cancer Institute; Michael Lawrence, Genentech Research and Early Development; and Levi Waldron, CUNY School of Public Health at Hunter College, New York.

The **Scientific Advisory Board** provides oversight through yearly meetings. Current members include: Simon Tavaré (Advisory Board chair; University of Southern California / Cambridge University); Robert Gentleman (23andMe); Paul Flicek (European Bioinformatics Institute); Simon Urbanek (AT&T Labs – Research); Wolfgang Huber (European Molecular Biology Laboratory); Vincent Carey (Brigham & Womens); Raphael Irizzary (Dana Farber).

4 Challenges and Opportunities

4.1 Project Relocation

The project will re-locate from Fred Hutchinson Cancer Research Center in Seattle, WA, USA to Roswell Park Cancer Institute (RPCI), Buffalo, NY, USA on 1 September 2015. All collaborators, relevant program officers, and Center administration are informed and supportive of this move. There is likely to be significant disruption during and after the move; several personnel with significant engagement in the project will not transition. Opportunities to mitigate disruption have been pursued, including transitioning employees who already work remotely, sub-contracting some work to FHCRC under the supervision of Dr. Raphael Gottardo, identifying new and talented employees at RPCI, and increasing reliance on generic cloud-based hardware infrastructure for our build system and web and support sites.

4.2 Cloud Computing

The 'download to desktop / user account' model places signficant burden on both users (struggling to install packages with idiosyncratic dependencies) and developers (needing to produce software that works across computing platforms). Additional issues involve data movement and storage. There are two directions implied by these considerations. The first involves continued elaboration of Docker and other portable software containers. The second involves use of these containers to deliver scalable computing facilities to appropriate segments of our user community, e.g., using the high-performance computing cluster at SUNY Buffalo.

4.3 Revision Control and Build Systems

Revision control (SVN) and the nightly build system are now far from state-of-the-art. We have developed a patchwork solution to support a git(hub) / svn bridge, but this fails with transforming git's non-linear commit histories to svn. It also creates confusion about the definitive repository for a package. A more comprehensive solution is needed. The solution probably involves transitioning to git (sans hub) as the primary archive, just as svn replaced cvs at an early stage in the project. There is some expertise within core team members, notably Dan Tenenbaum and James Hester, to facilitate this transition.

Our nightly build system followed by public availability of updated packages conflates package check and package distribution. This leads to an artificial lag between commit and testing, and distribution of packages with impossible-to-satisfy dependencies. Simply adopting (public) continuous integration systems like Travis do not address our cross-platform requirements, and do not adequately test the full consequence of commit changes for dependent packages. We require a re-engineered build system. There is relevant expertise that we can leverage to address this, e.g., Gabe Becker's efforts on switchr, as well as possible influence with any R consortium efforts at providing this type of service.

4.4 Project Participation

There are three dimensions of project participation that represent growing points. The project's key strengths derive from its appeal to separate communities: statistics; computing; and biology.

The first challenge is to remain accessible to, and rewarding for, each of these communities, so that new package contributions remain on the leading edge of bioinformatics.

The second challenge reflects the diversity of domain areas in which *Bioconductor* has credible strength. Differential expression represents a primary focus, but there is considerable expertise in other areas of high throughput sequencing, as well as significant contributions in flow cytometry and proteomics. More generally there are *R* communities (e.g., ecology, phylogeny, rOpenSci) which offer the opportunity for considerable synergy. The challenge then is to engage and nurture these domains of expertise. Approaches include focused activities (e.g.,

facilitating flow or proteomics workshops), active engagement in relevant communities (e.g., advisory board members overlapping rOpenSci?), and traditional scientific grantsmanship (e.g., letters of support or collaborative proposals).

The third challenge involves transitions from user to developer, and from developer to thought leader. The latter transition is a particularly valuable opportunity, as evident at the *Bioconductor* annual conference where there were an intimidating group of graduate students, post-docs, and junior faculty making valuable contributions to *Bioconductor*. How is this group's enthusiasm and contribution to be marshalled into long-term and productive commitment to *Bioconductor*?

4.5 Software needs

There are a number of directions for software development. A very incomplete list includes: Effective work with on-disk data resources. Coordinated multi-assay representations. Better BiocParallel engineering (memory management). Framework for interactive visualization.

4.6 Facile and Robust Package Development

R provides flexible facilities for creative statistical solutions, and it's package system provides sufficient structure to allow implementation and distribution of many novel analysis techniques. *Bioconductor* has augmented these facilities with emphasis on formal data structures for cross-package interoperability and reproducible research, end-user oriented vignettes, etc.

Some of the developments in *R* and *Bioconductor* have come at quite a considerable cost, even while improving overall software quality. The process of producing a usable package now represents a very considerable effort with increasingly demanding requirements for technical knowledge. The use of structured data in S4 classes often serves to bewilder both users (who become lost in the jungle of classes and the standards to which they are implemented) and developers (who 'know' the value of formal classes, but cannot identify appropriate reusable components for their needs). At the same time, increasingly complex computational demands and package dependencies demand adoption of more comprehensive software best practices ranging from consistent formatting and package structure to comprehensive unit testing. This places further demands on the technical skills of the developer, and erects an ever-higher barrier between new idea and usable implementation.

It is important to restrict the diversity of S4 classes the user deals with. One avenue toward this may be to emphasize general abstractions, rather than insist on strong type specification. A second approach might more aggressively address (or learn to live with) conceptual and other defects in base R data structures, rather than implementing a work-around that provides value only to intermediate and advanced users.

In many ways one would like to impose uniform coding expectations for *Bioconductor* packages, for instance adopting a single unit testing framework (e.g., RUnit versus testthat) and model for documentation (e.g., manually created 'man' pages versus roxygen markup). The challenges are that popular approaches are not always demonstrably better, that new approaches only gradually replace older approaches and the appropriate time to 'switch' is not always apparent, and that there is already considerable heterogeneity in the legacy code base.