

Bioc Technical Advisory Board Minutes

2 January 2025

Present: Vince Carey, Kasper Hansen, Helena Crowell, Jacques Serizay, Hervé Pagès, Alexandru Mahmoud, Lori Kern, Charlotte Soneson, Jen Wokaty, Rafael Irizarry, Sehyun Oh, Henrik Bengtsson, Stephanie Hicks, Marcel Ramos

Apologies: Wolfgang Huber, Laurent Gatto, Davide Risso, Levi Waldron, Ludwig Geistlinger, Michael Lawrence

:03 - :05 Previous [minutes](#) approved

:05 - :20 Update on U24 renewal

"In this proposal, the project will maintain continuity of availability of valued resources and processes, but also pivot strongly to take advantage of new techniques in information science and computing to modernize Bioconductor methods and resources and to strengthen translatability of discoveries in computational genomic data science to advancement of human health."

- How should we pivot?
 - Accelerate code and analysis processes with LLMs – apparently Bioc is already well known to Claude 3.5, let it build new clients for interrogation and integration of existing and new resources, etc.
 - How could we estimate cost?
 - We can spend a reasonable fraction of the budget on commodity solutions (e.g., RAG, chatbot production) so that team effort can be spent on unique uncommercialized aspects of the project, but we don't know how to estimate product costs over 5 years and have a tight bottom line.
 - Is there anything that can/should be done to a GitHub repository to make it more accessible to LLMs, easier to interface? We already have all packages on GitHub. What about GitLab?
 - Can we make a connection to literate programming? Write the description first, followed by code. LLMs need detailed instructions.
 - Would there be a way to write instructions to help with/automate unit test writing?
 - Replace gitolite and git with GitLab or GitHub (for authentication), automate more of BBS activities, collaborating with R universe.
 - R multiverse concepts also relevant – rethink the cadence, developer experience requirements.
 - Strong commitment to introspection, resource usage measurement.
 - Minimize overhead/work required to keep the build system running. Critical failure point for the project.
 - Emphasize containerization and availability of precompiled binaries
 - How can new mac and Windows containers affect developer experience?

- Accelerate discovery of interoperability problems, push this to the developer.
 - Document cost and environmental advantages of non-recompilation.
 - What are downsides of containerization? Security profile? Do practicing genomic data scientists take advantage of this?
- Language-agnostic/cross-language resources.
 - Examples of basilisk, alabaster, tatami/libscan/scrapper.
- Other topics
 - GPU, GA4GH and workflow concepts (WES, TRS, TES), unified API to hubs.

:20 - :25 Working group proposal from Sehyun Oh on image processing

- Description: NCI U24: Multi-omic Integration of Histopathology Image Analysis
- Chair: Sehyun Oh
- Initial Members: Mohamed Omar, Ilaria Billato, Hervé Pagès, Levi Waldron, Davide Risso, Chiara Romualdi, Luigi Marchionni, Marcel Ramos, Sean Davis, Lucio Queiroz
- Relevant links: [Meeting notes](#)
- Justification for necessity: This working group addresses the need for standardized workflows to integrate histopathology image-derived features with genomic and transcriptomic analyses in R/Bioconductor. We will develop data structures and workflows for processing image analysis outputs, create a curated repository of TCGA image features, and provide comprehensive training resources.
- Initial goals: For the next two years, we plan to establish standardized workflows for importing image features from Python-based tools into R/Bioconductor data structures, process TCGA histopathology images and create a curated repository of extracted features, and develop educational materials and documentation for integrated image-omic analysis.

:25 - :27 CAB update

- Had a nice recap of BiocAsia.
- Discussed the potential impact of the EU Cyber Resilience Act.
- New Working group “Event/Engagement” (name a work in progress) for increasing events, workshops, conferences.

:27 - :31 Bioconductor and GPUs

- Experiences from ITCR/CZI projects.
- How does the hovernet application interact with Bioconductor (Levi, Hervé, Davide)? Are there new data containers for annotated images - can interleave with Bioc and LLMs
- Comments on interests in harvesting data from Bioconductor doc and support content (Stephanie, Ludwig)? RAG and chatbot development via commodity platforms? [Autonomous agents](#) for analytic strategies?
 - Bioconductor chatbot would democratize access to coding and analysis support (especially for users who may be hesitant to ask questions on public forums).
 - Several solutions have been discussed:

- Migration of support site to Discourse that comes with integrated AI support.
 - A minimal shiny app that runs a generalist model in the background (as e.g. for Posit's Shiny Assistant), would require some funds.
 - Deployment of a fine-tuned RAG model, e.g. through Open WebUI or BioChatter - investigate whether specialist models outperform generalist ones.
 - Dedicated slack channel may be useful (#generative-ai or #llm)
- Connect with Maria/Aedin related to website search functionality improvement.

:31 - :39 Deliverables from conferences - what's presented/accomplished from a tech contribution point of view?

- Does it have to be prescribed or does it emerge organically?
- Embed hackathon concepts into conferences.
- CodeFest.
- Define deliverables in terms of attendance, first-time developers.
- Attendee survey as a measure of conference quality.
- Consider a BiocConferences package with data, there are google drives with info.
- Get more information from workshop platform (number of launches, uptime, can look for specific files to infer whether the workshop was run up to a specific point, profiling).
- Hybrid conferences - reach more participants, platform cost.

:39 - :41 Workflows

- Looking for alternatives to F1000Research
- New workflow from Sehyun: <https://f1000research.com/articles/13-1257>
- Another innovative publishing model: <https://www.reviewcommons.org/authors/>
- Tension between the “[version of record](#)” concept in publishing versus the dynamic, frequent-updates culture of R/CRAN/Bioc. Also, reviews in practice tend to refer to one particular version of the work.

:41 - :43 Current thinking about build system renovation for software packages.

- Transition hosting process to GitLab (drop gitolite) on commodity cloud hardware for availability.
- Check/report/propagate via GitHub Actions in collaboration with r-universe.
- Use r-universe API to develop analogs to "build reports".
- Distribution of binaries as GitHub packages.
- Possible role of r-multiverse.

:43 - :60 Open floor.

- Explorations of S7
 - Michael Lawrence will demo S7 in an upcoming meeting of the biocclasses working group.
 - Before migrating away from S4, need high test coverage to make sure nothing is breaking.

- Parallel testing - BiocFileCache causes issues (corrupt cache). Initial attempt with locking mechanism does not seem to be enough. Reproducible example would be helpful.
 - Microsoft DataLake is the current place for user-contributed Hub data. Users are given access to a staging area where contributions are being reviewed. Access with SaaS tokens is being removed, and thus we need a new staging location -> new Hub ingester is in development. Dynamic endpoint, temporary credentials. Based on versity. In addition, want to encourage contributors to host files on e.g. zenodo, CloudFlare, own S3 buckets etc (can still be accessed via ExperimentHub). Hosting data on e.g. zenodo makes contributors more independent, reduces work from core, may promote submission/update of more data.
-

Appendix (Levi): Draft requirements with AI.

"Can you draft a set of specifications for any replacement of the Bioconductor build system?"

Sure! Here are some specifications for a replacement of the Bioconductor Build System (BBS):

1. System Overview

- Purpose: To automate the building, testing, and reporting of Bioconductor packages across multiple platforms.
- Platforms Supported: Linux (x8664, aarch64), macOS (x8664, ARM64), and Windows.

2. Build Process

- Continuous Integration: Implement a continuous integration system to replace the nightly build system. This could be achieved using tools like GitHub Actions, Jenkins, or GitLab CI.
- Incremental Builds: Support for incremental builds to reduce build times by only rebuilding packages that have changed.
- Parallel Builds: Utilize parallel processing to build multiple packages simultaneously.

3. Testing and Validation

- Automated Testing: Integrate automated testing frameworks to run unit tests, integration tests, and system tests.
- Package Checks: Ensure packages pass R CMD check and BiocCheck[1].
- Dependency Management: Automatically resolve and install package dependencies.

4. Reporting and Notifications

- Build Reports: Generate detailed build reports, including logs of successful and failed builds.
- Notifications: Send notifications to package maintainers about build status via email or other communication channels.
- Dashboard: Provide a web-based dashboard for monitoring build status and accessing logs.

5. Infrastructure and Scalability

- Cloud Integration: Leverage cloud services (e.g., AWS, Azure) for scalable and flexible build environments.

- Containerization: Use Docker or similar containerization technologies to ensure consistent build environments across different platforms.
- Resource Management: Implement resource management to optimize the use of build machines and reduce costs.

6. Security and Compliance

- Access Control: Implement robust access control mechanisms to secure the build system.
- Data Privacy: Ensure compliance with data privacy regulations and best practices.
- Backup and Recovery: Regularly back up build data and implement disaster recovery plans.

7. Documentation and Support

- User Documentation: Provide comprehensive documentation for users, including installation guides, usage instructions, and troubleshooting tips.
- Developer Documentation: Maintain detailed documentation for developers contributing to the build system.
- Support Channels: Establish support channels for users and developers to seek help and report issues.

These specifications aim to modernize the Bioconductor build system, making it more efficient, scalable, and user-friendly. If you have any specific requirements or additional features in mind, feel free to let me know!

References

[1] [BiocCheck: Ensuring Bioconductor package guidelines](#)