

Xpress-Bio: a general purpose NodeJS based buildless server framework oriented towards bioinformatics applications

23 June 2025

Authors:

Ibrahim Tanyalcin^{*†}, Rodrigo Pacifico^{††}, Shaida Moghaddassi^{††}, Nicholas Hand^{††},
Alex Cherekos^{†††}, Rui Portela[†], Sébastien Janas[†]

* Corresponding author [†] GSK, Rixensaart, Belgium ^{††} GSK, Cambridge, MS,
US ^{†††} GSK, Upper Providence, PA, US

Summary

Xpress-Bio is a buildless, extendible and optionally containerized web server application for interactive genomic data exploration and analysis. It combines widely used bioinformatics tools (e.g., Samtools, BLAST+, HTSlib, IGV.js) into a unified framework that supports web technologies such as Server-Sent Events, WebSockets and subscriptions in the frontend. In the backend, it takes advantage of multi-threaded worker patterns implemented in NodeJS. Designed for bioinformaticians who prefer low-abstraction, high control setups, Xpress-Bio operates on both standalone and containerized environments (Docker, Singularity) with a focus on ease of extension, reproducibility, and control over data processing.

Statement of need

While platforms like Galaxy and EMBOSS provide general-purpose bioinformatics functionality, they impose rigid templates that limit custom routing, extensibility, and real-time interaction (Rice, Longden, and Bleasby 2000, Afgan et al. (2018)). Xpress-Bio offers a lower-level, developer-centric alternative that still includes critical functionality such as live data updates, BLAST search, and indexed genome browsing.

Unlike typical web frameworks, Xpress-Bio is pre-configured for genomics workflows out of the box. It handles automatic indexing (.fai, .bai, .csi, .tbi), dynamic BLAST database generation, and efficient cache management. The framework exposes Express.js route configuration, server state, event emitters, and WebComponent-based UI elements that allow full-stack extensibility.

Xpress-Bio uses a modular architecture for server routes. Route modules placed under the **routes/** directory are automatically discovered and loaded at startup. Each file can define one or more routes in the form of sync/async functions that receive shared utility objects as arguments. Hot reloading is supported via nodemon in the form of full server restart. During full restarts, all active worker threads are gracefully terminated and clients reconnect automatically on the frontend. Resource intensive or long running tasks are executed via persistent worker threads that support bidirectional communication via **MessageChannels**.

All in all, Xpress-Bio is suited to research scientists that need fast, no-build, containerized BLAST, visualization, and LLM stacks that need real-time communications with server-sent events or web-sockets.

Implementation and Features

Implementation

Built upon Express.js/Node.js, requires no build step to initialize or start the server. New routes are a collection of functions that can be dropped under **app/js/server/routes**. These are plain js files that define a single function which receives **express**, **app**, **info**, **files**, **serverSent**, **ws** and **cache** parameters. **Express** is the express js constructor, **app** is the app instance, **info** is server information with all custom data as a result of merged config files, **files** are list of files that can be requested by clients, **serverSent** and **ws** are **proxy** objects that can be utilized to send individual or broadcast messages via server-sent events or web-sockets and finally **cache** is the memcached key-value storage with 1 month expiry.

BLAST support: Queries run on dedicated threads using a batch scheduler and return results in both standard and tabular formats [Camacho et al. (2009);]. (see Figure 1, panels c–f)

UI built using native WebComponents and IGV.js, enabling multi-track visualizations and real-time interaction [Robinson et al. (2011);].

Features

Works with FASTA, BAM, GFF files and supports automated indexing via Samtools/HTSlib [Bonfield et al. (2021);]. (see Figure 1, panels c–f)

Users can run BLAST on the uploaded sequences. BLAST results are generated in 2 formats and are downloadable. Clicking on BLAST results navigates to the

corresponding locus within the selected IGV applet.

Supports Server-Sent Events for real-time updates and WebSockets for bi-directional communication. (see Figure 1, panels a, b, g). The server maintains responsiveness up to 100 concurrent clients.

LLM integration: Users can connect public/private LLMs to sequence data via a tool called “g-nome”, enabling conversational querying and UI extension through proposed code snippets. (see Figure 1, panel g). Multiple agents can be spawned simultaneously. Agents do not run code automatically, the user has the opportunity to review the snippets inside the chat component before they are executed. Data sent includes HTML tree, visible sequence, filenames and other metadata such as BLAST content. These can be tweaked or removed by the developers if necessary.

Example: BLAST Query Performance

Queries were performed against the **Chinese hamster genome** derivative, a fairly large FASTA file as reference

Table 1. BLAST query performance measured on a 6-core Ryzen 5 PRO system.

Number of queries	Query length (nt)	Execution time (s)
1-5	20-30	< 5
5-10	20-30	5-10
10-20	20-30	5-15

Measured on a 6-core, 12-thread Ryzen 5 PRO with 16GB RAM. Queries were performed on Chinese hamster genome*. Queries and database generation are processed in separate worker threads for responsiveness.

* For comparisons, the size of the input genome is about 2.5Gb.

Figures

Acknowledgements

I.T. wrote the software, A.C reviewed pull-requests. N.H, Ro.P and S.M tested functionality and proposed routinely used features. Ro.P, S.M, Ru.P and S.J reviewed the manuscript and proposed features.

Funding

This work was sponsored by GlaxoSmithKline Biologicals SA.

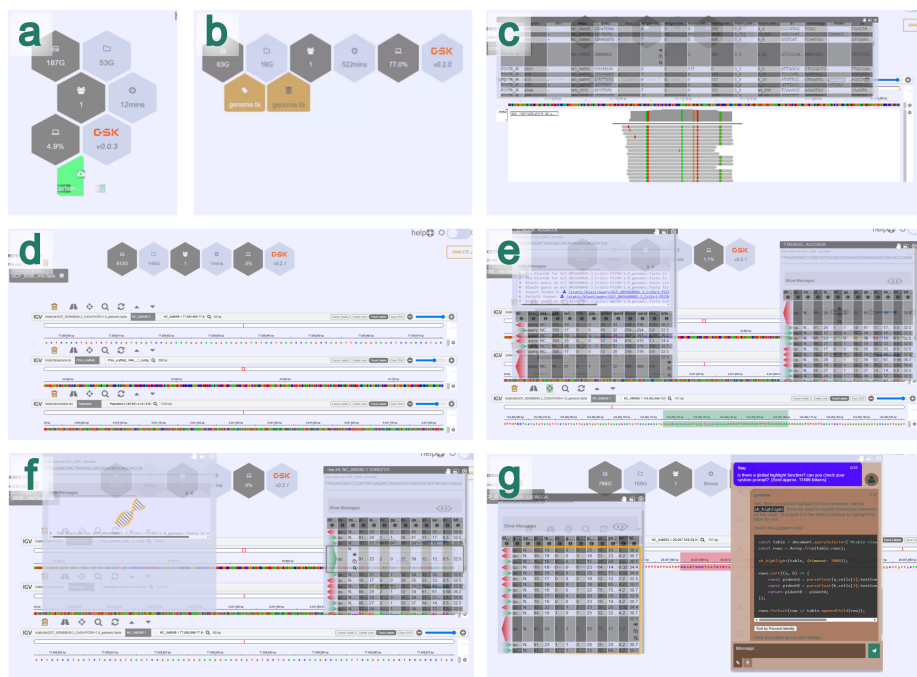


Figure 1: A non-exhaustive overview of Xpress-Bio features.

- (a) Uploading from URL or local file shows progress as a green hexagon.
- (b) Uploaded annotations, FASTA files, BAM files and BLAST databases are indexed or created on separate threads simultaneously. The progress is shown as pulsating orange hexagons.
- (c) Tables can be joined with FASTA/BAM pairs. Clicking on table row that has position information relocates the IGV viewer to that location.
- (d) Multiple IGV applets can be added and reordered at any time. Each applet can load its own set of BAM files and annotations.
- (e) BLAST results render outputs in a draggable table. Each table shows the orientation (forward/reverse) of the hit in the first column. Multiple BLAST windows can be simultaneously opened and each BLAST result can be paired with one of IGV applets, enabling click-to-navigate behavior.
- (f) Users can initiate multiple BLAST queries simultaneously. The BLAST search takes place on a separate thread and utilizes available CPU cores. Results are cached for retrieval.
- (g) Users can open the “g-nome” app on the side bar to connect to public LLM models with their API keys. Models get access to currently visible sequence data along with any BLAST windows and tracks. Xpress-Bio provides a mechanism for these LLM models to propose code snippets that can be run by the client which allows manipulation of visual elements on the screen. This allows scientists to extend UI functionality like sorting BLAST outputs and searching keywords within files.

Conflict of interest

All authors are, or were at the time of the study employees of the GSK group of companies.

References

`./paper.bib`.

Afgan, Enis, Dannon Baker, Bérénice Batut, Marius van den Beek, Dave Bouvier, Martin Čech, John Chilton, et al. 2018. “The Galaxy Platform for Accessible, Reproducible and Collaborative Biomedical Analyses: 2018 Update.” *Nucleic Acids Research* 46 (W1): W537–W544. doi:10.1093/nar/gky379.

Bonfield, James K, Joanne Marshall, Petr Danecek, Heng Li, Valtteri Ohan, Andrew Whitwham, Thomas Keane, Richard M Davies, and Greg Tischler. 2021. “HTSlib: C Library for Reading/Writing High-Throughput Sequencing Data.” *GigaScience* 10 (2): giab007. doi:10.1093/gigascience/giab007.

Camacho, Christiam, George Coulouris, Vahram Avagyan, Ning Ma, Jason Papadopoulos, Kevin Bealer, and Thomas L Madden. 2009. “BLAST+: Architecture and Applications.” *BMC Bioinformatics* 10 (1): 421. doi:10.1186/1471-2105-10-421.

Rice, Peter, Ian Longden, and Alan Bleasby. 2000. “EMBOSS: The European Molecular Biology Open Software Suite.” *Trends in Genetics* 16 (6): 276–77. doi:10.1016/S0168-9525(00)02024-2.

Robinson, James T., Helga Thorvaldsdóttir, Wendy Winckler, Mitchell Guttman, Eric S. Lander, Gad Getz, and Jill P. Mesirov. 2011. “Integrative Genomics Viewer.” *Nature Biotechnology* 29 (January). Nature Publishing Group, a division of Macmillan Publishers Limited. All Rights Reserved. SN -: 24–26. doi:10.1038/nbt.1754.