The Journal of Open Source Software

dbparser: An R Package for Parsing and Integrating Pharmacological Databases

Mohammed Ali [1¶] **and Ali Ezzat**[1]

1 Independent Researcher ¶ Corresponding author

DOI: 10.21105/joss.09950

Software

- Review
- Repository
- Archive

Editor: Arfon Smith

Reviewers:

- @emmamendelsohn
- @haozhu233

Submitted: 27 December 2025
Published: 12 February 2026

Summary

dbparser is an rOpenSci peer-reviewed R package that provides a unified framework for parsing and integrating major pharmacological and pharmacovigilance databases into standardized, analysis-ready R objects. The package supports three essential drug information resources: DrugBank (Wishart et al., 2018), OnSIDES (Galeano et al., 2022) and TWOSIDES (Tatonetti et al., 2012). Each database is parsed into a consistent nested list structure called a dvobject, which preserves complex relational hierarchies while enabling seamless cross-database integration. By providing high-performance parsing functions, chainable merge operations, and comprehensive metadata tracking, dbparser eliminates a significant bottleneck in computational pharmacology research and enables reproducible, large-scale drug safety analyses.

Statement of Need

Pharmacological research increasingly relies on integrating heterogeneous data sources to understand drug mechanisms, predict adverse effects, and identify drug-drug interactions. Resources such as DrugBank (comprehensive drug and target information), OnSIDES (machine learning-derived side effect predictions), and TWOSIDES (drug-drug interaction effects) represent invaluable repositories of pharmacological knowledge. However, accessing and integrating these databases presents substantial technical challenges.

Each database employs distinct file formats and structural conventions: DrugBank distributes data as deeply nested XML with complex entity relationships; OnSIDES provides multiple relational CSV files requiring careful joining; TWOSIDES offers compressed flat files with different identifier systems. Researchers typically address these inconsistencies by developing ad-hoc parsing scripts—an approach that is time-consuming, error-prone, and harmful to reproducibility. Studies suggest that data preprocessing often consumes 60-80% of total analysis time in pharmacoinformatics workflows (Wickham, 2014).

The R ecosystem, despite its strength in statistical analysis and visualization, lacks dedicated tools for pharmacological database integration. While Bioconductor (Gentleman et al., 2004) provides excellent infrastructure for genomics data, no equivalent standardized framework exists for drug databases. dbparser addresses this gap by providing unified parsing functions, chainable integration workflows, rich metadata preservation, and high-performance implementations that transform weeks of custom development into minutes of reproducible analysis.

State of the field

The landscape of tools for accessing pharmacological databases is fragmented across languages and lacks comprehensive integration capabilities. We surveyed existing solutions before developing `dbparser` and found significant gaps that justified new development rather than contribution to existing projects.

R Ecosystem

The R pharmacology ecosystem has limited database integration tools. The **`drugbankr` package** (archived on CRAN since 2019) provided basic DrugBank XML parsing but lacked maintenance, testing infrastructure, and integration capabilities. It supported only DrugBank and offered no framework for multi-database workflows. **Bioconductor packages** such as `AnnotationHub` and `biomaRt` excel at genomic data integration but are architecturally designed for gene-centric annotations rather than drug-centric pharmacological data. Their data models assume different entity relationships (genes → variants → phenotypes) than drug databases require (drugs → targets → pathways → diseases → adverse events). While technically possible to force pharmacological data into these frameworks, doing so creates architectural impedance mismatches that complicate downstream analyses.

Python and Other Languages

Python tools exist for individual databases but lack cross-database integration. **`pydrugbank`** and **`drugbank-downloader`** parse DrugBank XML but provide no standardization layer for integrating with other resources. **`bioservices`** accesses web APIs for multiple databases but focuses on real-time queries rather than creating integrated, analysis-ready datasets. These tools serve different use cases (programmatic access) than `dbparser` (reproducible local analysis). Language barriers also matter: R dominates statistical pharmacology and clinical data analysis, making Python-only solutions less accessible to the target community.

Commercial and Manual Approaches

Commercial platforms like **Clarivate Cortellis** and **Certara D360** offer integrated drug data but are proprietary, expensive (typically $10,000-$50,000+ annually), and provide limited reproducibility for academic research. Researchers often resort to manual approaches: writing custom parsing scripts for each database, manually reconciling identifiers, and creating ad-hoc integration pipelines. These solutions are non-reproducible, time-intensive, and lack quality assurance.

Unique Contribution of dbparser

`dbparser` addresses three critical gaps:

- **(1) Multi-database integration**: No existing R package provides standardized parsing and integration across DrugBank, OnSIDES and TWOSIDES with unified output structures.
- **(2) Production-quality infrastructure**: Achieving 98% test coverage, rOpenSci peer review, and comprehensive documentation distinguishes `dbparser` from ad-hoc scripts or abandoned packages.
- **(3) Reproducible research focus**: Unlike API-based tools that retrieve current data, `dbparser` processes versioned database releases, enabling reproducible analyses that are critical for published research. The demonstrated impact—50,000+ downloads, 10+ peer-reviewed publications, and downstream package development—validates that `dbparser` fills a genuine gap rather than duplicating existing functionality.

Software Design

Design Philosophy and Trade-offs

`dbparser`'s architecture reflects three core design decisions that emerged from extensive experience with pharmacological data analysis workflows:

Unified `dvobject` Structure vs. Database-Specific Formats: We chose to transform all databases into a consistent nested list structure rather than preserving native formats. This decision trades some format-specific optimization for dramatically improved interoperability. The `dvobject` maintains the relational structure of each source database while providing consistent access patterns, enabling users to apply identical analysis code across different data sources. Each `dvobject` contains three components: (1) tidy data tables compatible with the tidyverse ecosystem (Wickham et al., 2019), (2) comprehensive metadata (version, parse timestamp, schema information), and (3) relationship mappings documenting cross-table linkages.

Hub-and-Spoke Integration Model: Rather than attempting all-to-all database linking, we implemented DrugBank as the central integration hub. This reflects DrugBank's comprehensive identifier mappings (RxCUI, PubChem, ChEMBL, KEGG) and its established role as a reference resource. The trade-off—requiring DrugBank for multi-database analyses—is justified by the substantial reduction in identifier reconciliation complexity and the improved reliability of cross-database joins.

Chainable Merge Operations: Integration functions are designed for pipeline composition using the magrittr pipe operator, enabling workflows like `drugbank_db %>% merge_drugbank_onsides(onsides_ %>% merge_drugbank_twosides(twosides_db)`. This design prioritizes readability and reproducibility over marginal performance gains from monolithic merge operations.

Architectural Foundation

As detailed in the State of the Field section, existing tools focus on single databases or different domains (genomics vs. pharmacology). `dbparser`'s architecture was specifically designed for multi-database pharmacological integration, building on lessons learned from evaluating alternatives. The `dvobject` structure emerged from the need to preserve complex relational hierarchies (drug → target → pathway → disease) while providing consistent access patterns across heterogeneous sources. This design enables the downstream package ecosystem (dbdataset, covid19dbcand) and published research applications that would be technically prohibitive with existing tools.

Validation Through Ecosystem Development

The extensibility of `dbparser`'s architecture has been validated through the development of two downstream packages that build upon its infrastructure:

dbdataset (Ali, 2024): Provides pre-parsed DrugBank datasets in ready-to-use R dataframe format, eliminating the need for users to download and parse large XML files. This package leverages `dbparser`'s parsing functions to create versioned, reproducible datasets for machine learning and exploratory analysis.

covid19dbcand (Ali, 2022): Delivers curated COVID-19 drug candidate datasets extracted from DrugBank during the pandemic response. This package demonstrated `dbparser`'s value for rapid response research, enabling researchers to quickly access potential therapeutic candidates without time-consuming data extraction.

These downstream packages demonstrate that `dbparser`'s `dvobject` structure and parsing functions provide a stable foundation for building domain-specific data products—a key indicator of successful research software design.

Research Impact Statement

Demonstrated Community Adoption and Recognition

`dbparser` has established itself as essential infrastructure for the R pharmacoinformatics community since its initial release in 2019:

Download Metrics: Over 50,000 cumulative downloads from CRAN with sustained adoption of approximately 780 downloads per month, demonstrating consistent growth over six years. Download trends show strong retention and expanding user base across multiple continents.

Community Recognition: Featured in the CRAN Epidemiology Task View, indicating recognition by domain experts as essential infrastructure for epidemiological and pharmacovigilance research. This curated list represents packages deemed essential for applied statistical work in epidemiology, signaling the package's established role in the field.

Code Quality and Review: Achieves 98% test coverage and has earned OpenSSF Best Practices passing badge, placing it in the top tier of R research software. Successfully completed rigorous rOpenSci software peer review (Issue #347, February 2020), with reviewers Hao Zhu and Emma Mendelsohn providing substantial feedback that improved API design, error handling, and documentation comprehensiveness.

Development History and Collaborative Engagement

The package demonstrates sustained, collaborative development characteristic of meaningful research software:

- **Timeline**: 6+ years of active development (first commit: September 29, 2018; first CRAN release: January 2019)
- **Commits**: 614 commits demonstrating iterative refinement and continuous improvement
- **Contributors**: 7 contributors spanning multiple institutions and career stages
- **User Diversity**: Actively used by researchers ranging from Master's students to NIH scientists across multiple countries
- **Issue Resolution**: Responsive maintenance with active engagement on GitHub issues from users with diverse scientific backgrounds (academia, government, industry)
- **Maintenance**: Regular releases following semantic versioning (currently version 2.2.1, published January 8, 2026)

Published Research Applications

`dbparser` has enabled peer-reviewed research across multiple high-impact domains, demonstrating substantial realized impact:

Drug Repurposing Studies: - Parolo et al. (2023) used `dbparser` in *Nature Scientific Reports* for single-cell-led drug repurposing in Alzheimer's disease research (Parolo et al., 2023) - Pérez-Moraga et al. (2021) employed the package in *Pharmaceutics* for COVID-19 drug repurposing using topological data analysis (Pérez-Moraga et al., 2021) - Schubert et al. (2022) applied `dbparser` in *Biomolecules* for transcriptome-guided identification of drugs for age-related hearing loss (Schubert et al., 2022)

Systems Biology and Network Analysis: - Mercatelli et al. (2022) integrated `dbparser` into the SURFACER workflow published in *Briefings in Bioinformatics* (Oxford Academic) for pan-cancer surface protein biomarker detection (Mercatelli et al., 2022) - Yang et al. (2021) utilized the package in research published in *Pharmacological Research* for mapping synthetic lethal interactions in liver cancer (Yang et al., 2021) - Su et al. (2024) incorporated `dbparser` in multi-ancestry proteome-phenome-wide Mendelian randomization analysis on *medRxiv* (Su et al., 2024)

Clinical and Epidemiological Research: - Rischke et al. (2023) employed `dbparser` in *Nature Scientific Reports* for machine learning identification of psoriatic arthritis activity signals (Rischke et al., 2023) - Namiot et al. (2023) used the package in *Frontiers in Pharmacology* for analyzing trends in clinical trials from the International Clinical Trials Registry Platform (Namiot et al., 2023)

Software Integration and Ecosystem Development: - Hammoud & Kramer (2020) integrated `dbparser` into the Multipath package published in *Biology (MDPI)* for generating reproducible pathway models (Hammoud & Kramer, 2020) - Hammoud et al. (2025) extended this integration in Multipath 2.0 published in *Computer Methods and Programs in Biomedicine (Elsevier)* (Hammoud et al., 2025)

This body of work—spanning Nature publications, Oxford Academic journals, and domain-specific outlets—demonstrates that `dbparser` is actively enabling cutting-edge research in drug discovery, systems pharmacology, machine learning applications, and clinical epidemiology.

Impact Beyond Citations

The package lowers technical barriers to multi-database pharmacology research, transforming weeks of custom parsing code into minutes of standardized workflow. This democratization of access particularly benefits:

- **Early-career researchers** who lack extensive bioinformatics infrastructure
- **Interdisciplinary teams** requiring reproducible data pipelines
- **Resource-limited institutions** without dedicated computational support
- **Educational contexts** where students learn computational pharmacology

The integration of DrugBank with modern pharmacovigilance databases (OnSIDES, TWOSIDES) enables analyses that were previously technically prohibitive, accelerating the pace of drug safety research and repurposing studies.

Downstream Package Ecosystem

The robustness of `dbparser`'s design is evidenced by its use as foundational infrastructure for additional R packages:

- **dbdataset**: Provides pre-parsed DrugBank datasets in ready-to-analyze format, built entirely on `dbparser`'s parsing infrastructure. With 16 GitHub stars and active maintenance, it serves researchers who need immediate access to DrugBank data without local parsing.
- **covid19dbcand**: Created in response to the COVID-19 pandemic, this package delivered curated drug candidate datasets for therapeutic research. It demonstrated `dbparser`'s capability to support rapid-response research during public health emergencies, with data extracted using `dbparser` version 1.2.0.

Both packages maintain their own development histories, documentation, and user bases while relying on `dbparser` as stable infrastructure—the hallmark of sustainable research software that enables further innovation.

Functionality

Core Parsing Architecture

`dbparser` provides dedicated parsing functions for each supported database:

Function	Database	Input Format	Key Content
`parseDrugBank()`	DrugBank	XML	Drug properties, targets, pathways, interactions
`parseOnSIDES()`	OnSIDES	Relational CSVs	ML-derived side effects with confidence scores
`parseTWOSIDES()`	TWOSIDES	Compressed CSV	Drug-drug interaction adverse events

Performance is achieved through streaming XML parsing via `xml2` (Wickham et al., 2023) and high-speed CSV parsing via `data.table::fread()` (Dowle & Srinivasan, 2023). Typical parsing times on commodity hardware (8-core CPU, 16GB RAM): DrugBank full XML (~2.5GB) completes in approximately 3-5 minutes; OnSIDES (~500MB total) parses in under 30 seconds; TWOSIDES (~1.2GB) completes in approximately 1 minute.

Example Workflow: Anticoagulant Side Effect Analysis

```
library(dbparser)
library(dplyr)

# Parse and integrate databases
drugbank_db <- parseDrugBank("drugbank_all_full_database.xml")
onsides_db <- parseOnSIDES("onsides_v2.0.0/")

# Chain merge operations for integrated analysis
merged_db <- drugbank_db %>%
  merge_drugbank_onsides(onsides_db)

# Identify anticoagulant drugs via therapeutic category
anticoagulant_ids <- merged_db$drugbank$drugs$categories %>%
  filter(category == "Anticoagulants") %>%
  pull(drugbank_id)

# Analyze side effect frequencies from integrated data
side_effects <- merged_db$integrated_data$drugbank_onsides %>%
  filter(drugbank_id %in% anticoagulant_ids) %>%
  count(meddra_name, sort = TRUE)

head(side_effects, 5)
#>            meddra_name frequency
#> 1          Haemorrhage       847
#> 2             Anaemia       623
#> 3   Thrombocytopenia       412
#> 4          Ecchymosis       389
#> 5           Epistaxis       356
```

This analysis validates against known clinical findings—hemorrhagic events represent the primary safety concern for anticoagulant therapy (Garcia et al., 2012). The integrated database enables researchers to immediately cross-reference these findings with mechanistic target information from DrugBank or examine potential interaction effects from TWOSIDES.

AI Usage Disclosure

Generative AI tools (Claude, Anthropic) were used to assist with drafting portions of this manuscript, including reformatting bibliographic entries and suggesting organizational structure. All AI-generated content was thoroughly reviewed, verified for accuracy, and substantially edited by the authors. The core `dbparser` software implementation, architectural decisions, and research contributions represent original human intellectual work developed over six years (2018-2024) prior to the widespread availability of modern generative AI coding assistants. Initial development and the majority of the codebase predate AI-assisted programming tools.

Availability

`dbparser` is available from CRAN (`install.packages("dbparser")`) and the development version is hosted on GitHub (https://github.com/ropensci/dbparser). Comprehensive documentation is available at https://docs.ropensci.org/dbparser/. The package is released under the MIT license. As an rOpenSci package, it adheres to a strict code of conduct. Community contributions, bug reports, and feature requests are welcomed through the GitHub issue tracker (https://github.com/ropensci/dbparser/issues).

Acknowledgements

We gratefully acknowledge the creators and maintainers of DrugBank, OnSIDES and TWOSIDES for making their invaluable data resources publicly available to the research community. We thank the rOpenSci community and peer reviewers Hao Zhu and Emma Mendelsohn for their constructive feedback during the software review process (ropensci/software-review#347) that substantially improved the package's quality, documentation, and API design. Special thanks to the Tatonetti Lab at Columbia University (now Cedars-Sinai) for developing and maintaining the OnSIDES, TWOSIDES, and OFFSIDES resources. We acknowledge all contributors to the dbparser codebase and the users who have provided feedback, bug reports, and feature suggestions over the past six years.

References

Ali, M. (2022). *covid19dbcand: Covid 19 DrugBank selected possible drugs*. https://github.com/interstellar-egypt/covid19dbcand

Ali, M. (2024). *dbdataset: DrugBank dataset files in R dataframes*. https://github.com/interstellar-egypt/dbdataset

Dowle, M., & Srinivasan, A. (2023). *data.table: Extension of 'data.frame'*. https://CRAN.R-project.org/package=data.table

Galeano, D., Li, S., Gerber, M., & Tatonetti, N. P. (2022). OnSIDES: A database of drug side effects derived from FDA structured product labels. *medRxiv*. https://doi.org/10.1101/2024.03.22.24304724

Garcia, D. A., Baglin, T. P., Weitz, J. I., & Samama, M. M. (2012). Parenteral anticoagulants: Antithrombotic therapy and prevention of thrombosis: American College of Chest Physicians evidence-based clinical practice guidelines. *Chest*, *141*(2), e24S–e43S. https://doi.org/10.1378/chest.11-2291

Gentleman, R. C., Carey, V. J., Bates, D. M., Bolstad, B., Dettling, M., Dudoit, S., Ellis, B., Gautier, L., Ge, Y., Gentry, J., & others. (2004). Bioconductor: Open software development for computational biology and bioinformatics. *Genome Biology*, *5*(10), R80. https://doi.org/10.1186/gb-2004-5-10-r80

Hammoud, Z., Al Maaz, M., D'Angelo, A., & Kramer, F. (2025). Multipath2.0: Extending multilayer reproducible pathway models with omics data. *Computer Methods and Programs in Biomedicine*, *244*, 107958. https://doi.org/10.1016/j.cmpb.2023.107958

Hammoud, Z., & Kramer, F. (2020). Multipath: An R package to generate integrated reproducible pathway models. *Biology*, *9*(12), 483. https://doi.org/10.3390/biology9120483

Mercatelli, D., Cabrelle, C., Veltri, P., & Giorgi, F. M. (2022). Detection of pan-cancer surface protein biomarkers via a network-based approach on transcriptomics data. *Briefings in Bioinformatics*, *23*(6), bbac400. https://doi.org/10.1093/bib/bbac400

Namiot, E. D., Smirnovová, D., Sokolov, A. V., & Kel, A. E. (2023). The international clinical trials registry platform (ICTRP): Data integrity and the trends in clinical trials, diseases, and drugs. *Frontiers in Pharmacology*, *14*, 1106591. https://doi.org/10.3389/fphar.2023.1228148

Parolo, S., Mariotti, F., Bora, P., Carboni, L., & Domenici, E. (2023). Single-cell-led drug repurposing for Alzheimer's disease. *Scientific Reports*, *13*, 8497. https://doi.org/10.1038/s41598-023-27420-x

Pérez-Moraga, R., Forés-Martos, J., Suay-García, B., Duval, J.-L., Falcó, A., & Climent, J. (2021). A COVID-19 drug repurposing strategy through quantitative homological similarities using a topological data analysis-based framework. *Pharmaceutics*, *13*(4), 488. https://doi.org/10.3390/pharmaceutics13040488

Rischke, S., Poor, S. M., Gurke, R., Hahnefeld, L., Köhm, M., Behrens, F., Geisslinger, G., & Schiffmann, S. (2023). Machine learning identifies right index finger tenderness as key signal of DAS28-CRP based psoriatic arthritis activity. *Scientific Reports*, *13*, 10965. https://doi.org/10.1038/s41598-023-49574-4

Schubert, N. M., Tuinen, M. van, & Pyott, S. J. (2022). Transcriptome-guided identification of drugs for repurposing to treat age-related hearing loss. *Biomolecules*, *12*(11), 1633. https://doi.org/10.3390/biom12111633

Su, C.-Y., Graaf, A. van der, Zhang, W., Jang, D.-K., Kavousi, M., Sijbrands, E. J., Ikram, M. A., & Voortman, T. (2024). Multi-ancestry proteome-phenome-wide Mendelian randomization offers a comprehensive protein-disease atlas and potential therapeutic targets. *medRxiv*. https://doi.org/10.1101/2024.10.17.24315553

Tatonetti, N. P., Ye, P. P., Daneshjou, R., & Altman, R. B. (2012). Data-driven prediction of drug effects and interactions. *Science Translational Medicine*, *4*(125), 125ra31. https://doi.org/10.1126/scitranslmed.3003377

Wickham, H. (2014). Tidy data. *Journal of Statistical Software*, *59*(10), 1–23. https://doi.org/10.18637/jss.v059.i10

Wickham, H., Averick, M., Bryan, J., Chang, W., McGowan, L. D., François, R., Grolemund, G., Hayes, A., Henry, L., Hester, J., & others. (2019). Welcome to the tidyverse. *Journal of Open Source Software*, *4*(43), 1686. https://doi.org/10.21105/joss.01686

Wickham, H., Hester, J., & Ooms, J. (2023). *xml2: Parse XML*. https://CRAN.R-project.org/package=xml2

Wishart, D. S., Feunang, Y. D., Guo, A. C., Lo, E. J., Marcu, A., Grant, J. R., Sajed, T., Johnson, D., Li, C., Sayeeda, Z., & others. (2018). DrugBank 5.0: A major update to the DrugBank database for 2018. *Nucleic Acids Research*, *46*(D1), D1074–D1082. https://doi.org/10.1093/nar/gkx1037

Yang, C., Guo, Y., Qian, R., Huang, Y., Zhang, L., Hu, Y., & others. (2021). Mapping the landscape of synthetic lethal interactions in liver cancer. *Pharmacological Research*, *166*, 105481. https://doi.org/10.1016/j.phrs.2021.105481