



tendibility of the `bupaR` ecosystem, we have revised the `s3` class inheritance of `log` objects. Fig. 2 visualises the new class inheritance schema. Both `eventlog` and `activitylog` are inherited from the new base `log` class, which in turn uses a `tbl_df` from the `tibble` package [10] as back-end data storage. When grouping is applied to a `log` class using the `group_by()` functions, it becomes a `grouped_log` to signify the presence of grouping variable(s).

### B. Augmenting Logs

As of `edeR` 0.9.0, our package for exploratory and descriptive event data analysis, all `append` and `append_column` arguments of descriptive metrics (e.g. `activity_frequency()`, `processing_time()`, etc.) have been deprecated in favour of a new `augment()` method, which is consistent with the `broom` package [10] for adding outputs of predictions and estimations to data. The new workflow is visualised in Fig. 3, and a code example is provided in Listing 1. For instance, we can calculate the throughput times for each case on the `sepsis` log and add these times back to the `sepsis` log as a new column "case\_throughput\_time".



Figure 3. Augmenting a log [7].

```

1 sepsis %>%
2   throughput_time(level = "case") %>%
3   augment(log = sepsis, columns = "throughput_time",
  ↪ prefix = "case")
  
```

Listing 1. R example of augmenting a log.

This new workflow ensures consistent separation between the outputs of descriptive metrics and `log` objects. Furthermore, the `augment()` method provides a standardised, flexible, and transparent way to enrich logs with descriptive metrics.

### C. Improved Data Manipulation

Significant changes have been made to the supported `dplyr` [10] methods for data manipulation in `bupaR` (e.g. `filter`, `mutate`, `slice`, etc.), most significantly to `group_by()`, for grouping event data for descriptive analyses. For example, the number of cases in which each activity was executed can be calculated using the code shown on line 1 in Listing 2.

```

1 sepsis %>% group_by(activity) %>% n_cases()
2 sepsis %>% group_by_ids(activity_id) %>% n_cases()
3 sepsis %>% group_by_activity() %>% n_cases()
  
```

Listing 2. R example of `group_by`.

A more convenient way of grouping `log` objects as of `bupaR` 0.5.0 is by using the `group_by_ids()` method, completed with the desired `bupaR` attribute function(s) (e.g. `activity_id`, `case_id`, etc.), or by directly using `group_by_activity()`, as shown on lines 2 and 3, respectively. These new grouping methods allow conducting grouped descriptive analyses more conveniently without knowing the underlying column names. Moreover, the handling of grouped logs is improved so that any metric can now be computed for any (set of) grouping variable(s).

## III. MATURITY & USAGE

Since its conception, `bupaR` has received over 800K downloads in over 160 countries. Users come from various industries, e.g., healthcare, governance, automotive, and academics. Stable versions of `bupaR` and other `bupaverse` packages can be installed from CRAN using `install.packages("bupaverse")` or, for the version with the latest patches and bugfixes, directly from GitHub<sup>1</sup> using `devtools::install_github("bupaverse/bupaverse")`. A demonstration of the release can be found here.<sup>2</sup> Furthermore, the `bupar.net` website contains ample documentation and examples on `bupaR` and the `bupaverse` packages.

## IV. CONCLUSION & FUTURE WORK

This paper presented the release highlights of `bupaR` 0.5.0, most notably the introduction of the activity log, a new standardised way to augment logs, and improved data manipulation.

Future releases will focus on extending the `bupaverse` ecosystem with new functionalities for process analysis and maintenance of existing code. New functionalities, such as Performance Spectrum [11], trace and activity clustering, social network mining and process discovery, are currently on the roadmap. Other functionalities can be requested using GitHub Issues.<sup>1</sup>

## ACKNOWLEDGMENT

The authors would like to warmly thank all users who are actively contributing to the `bupaR`-framework by submitting issues and pull requests on the GitHub<sup>1</sup> repositories.

## REFERENCES

- [1] B. van Dongen, A. de Medeiros, E. Verbeek, A. Weijters, and W. van der Aalst, "The ProM Framework: A New Era in Process Mining Tool Support," ser. LNCS, vol. 3536. Springer, 2005, pp. 444–454.
- [2] A. Berti, S. van Zelst, and W. van der Aalst, "Process Mining for Python (PM4Py): Bridging the Gap Between Process- and Data Science," ser. CEUR-WS, vol. 2374, 2019, pp. 13–16.
- [3] M. La Rosa, H. Reijers, W. van der Aalst, R. Dijkman, J. Mendling, M. Dumas *et al.*, "APROMORE: An Advanced Process Model Repository," *Expert Syst. Appl.*, vol. 38, no. 6, pp. 7029–7040, 2011.
- [4] G. Janssenswillen, B. Depaire, M. Swennen, M. Jans, and K. Vanhoof, "bupaR: Enabling Reproducible Business Process Analysis," *Knowl. Based Syst.*, vol. 163, pp. 927–930, 2019.
- [5] R Core Team, "R: A Language and Environment for Statistical Computing," 2022. [Online]. Available: <https://www.R-project.org>
- [6] G. Janssenswillen, F. Mannhardt, M. Creemers, B. Depaire, L. Jooker, N. Martin, and G. Van Houdt, "Extensions to the bupaR Ecosystem: An Overview," ser. CEUR-WS, vol. 2703, 2020, pp. 43–46.
- [7] "bupaR 0.5.0: What's new?" 2022. [Online]. Available: <https://bupar.net/2022/07/27/bupar-0-5-0-whats-new/>
- [8] N. Martin, G. Van Houdt, and G. Janssenswillen, "DaQAPO: Supporting Flexible and Fine-Grained Event Log Quality Assessment," *Expert Syst. Appl.*, vol. 191, p. 116274, 2022.
- [9] L. Bouarfa and J. Dankelman, "Workflow Mining and Outlier Detection from Clinical Activity Logs," *J. Biomed. Inform.*, vol. 45, no. 6, pp. 1185–1190, 2012.
- [10] H. Wickham, M. Averick, J. Bryan, W. Chang, L. McGowan, R. François *et al.*, "Welcome to the Tidyverse," *J. Open Source Softw.*, vol. 4, no. 43, p. 1686, 2019.
- [11] V. Denisov, E. Belkina, D. Fahland, and W. van der Aalst, "The Performance Spectrum Miner: Visual Analytics for Fine-Grained Performance Analysis of Processes," ser. CEUR-WS, vol. 2196, 2018, pp. 96–100.

<sup>1</sup><https://github.com/bupaverse/>

<sup>2</sup><https://tinyurl.com/icpmdemobupar>