

A beginner-friendly environment for exploring error messages in the Clojure programming language.

Tristan Kalvoda, Elena Machkasova, Jaydon Stanislawski, and John Walbran

University of Minnesota, Morris

Midwest Instruction and Computing Symposium, April 2025

# Outline

- 1 Overview of Clojure and Its Error Messages
- 2 Babel project
  - Setup and Goals
  - Exceptions Processing
- 3 Morse Viewers
- 4 Current State of the Project and Future Work

# Tristan's section

# Jaydon's section

# Jaydon's section

# John's section

John's section

# Current State of the Project

- Most of the work this year was spent structuring things for integration with Morse viewers.
- The introduction of the error labelling and prototyping this was pivotal in enabling data formatting.
- We currently have a small number of error messages labelled for demonstration purposes.

# Current State of the Project

- Most of the work this year was spent structuring things for integration with Morse viewers.
- The introduction of the error labelling and prototyping this was pivotal in enabling data formatting.
- We currently have a small number of error messages labelled for demonstration purposes.



# Current State of the Project

- Most of the work this year was spent structuring things for integration with Morse viewers.
- The introduction of the error labelling and prototyping this was pivotal in enabling data formatting.
- We currently have a small number of error messages labelled for demonstration purposes.

# Future Work

The following are areas of active development:

- Expand data labelling to all Babel error messages.
- Add hover text for specific terms to add definitions and supplementary information to the presented error message.
- Refining the end user work flow between working code and erroring code.

# Future Work

The following are areas of active development:

- Expand data labelling to all Babel error messages.
- Add hover text for specific terms to add definitions and supplementary information to the presented error message.
- Refining the end user work flow between working code and erroring code.

# Future Work

The following are areas of active development:

- Expand data labelling to all Babel error messages.
- Add hover text for specific terms to add definitions and supplementary information to the presented error message.
- Refining the end user work flow between working code and erroring code.

## Future Work (cont.)

- Once we have greater feature coverage in Babel, we plan to run a usability study about the interactive tools we have developed.
- We are going to use the results of letting users explore our tools while learning Clojure in order to guide further design.
- We would like to explore IDE integration to further expand possible work-flow refinements.

## Future Work (cont.)

- Once we have greater feature coverage in Babel, we plan to run a usability study about the interactive tools we have developed.
- We are going to use the results of letting users explore our tools while learning Clojure in order to guide further design.
- We would like to explore IDE integration to further expand possible work-flow refinements.

## Future Work (cont.)

- Once we have greater feature coverage in Babel, we plan to run a usability study about the interactive tools we have developed.
- We are going to use the results of letting users explore our tools while learning Clojure in order to guide further design.
- We would like to explore IDE integration to further expand possible work-flow refinements.

# Acknowledgements

We would like to thank Joe Lane for introducing us to Morse tools and for numerous helpful discussions.



# Discussion

Questions?