

# Intro to TLA+

Jan Pokorný

SW engineer @ Red Hat

*RHEL High Availability Add-On*

**poki**

(nick seldom left  
available  
at various sites,  
but nonetheless)

# What's TLA+

- Temporal **L**ogic of **A**ctions
- Formal specification language
- Practical, model not just a blueprint (design), but alike...
- ... concise documentation for some traits of the system  
(if you can understand it, ASCII notation can be math-pretty-printed)
- ... subject of formal verification: **model checking** (exhaustive testing)
  - does the property hold in the specification?
  - **safety** & liveness
- *Relevancy: well-suited for distributed systems*

# What not to expect today

- TLA Toolbox in demos
  - let's proceed in geek's mode: just old good terminal
  - GUI/IDE shall be straightforward to grasp, and is taught in the official video course, see the references at the end
- PlusCal
  - imperative pseudocode translated into TLA+, anyway
  - let's proceed from the grounds up (assuming some pre-existing math knowledge...), can be a future topic :-)
- TLAPS (Proof System), ditto :-)
- Distributed systems per se :-/

# Author of TLA+

- *Let me reuse pre-existing slides :-)*
  - *[https://github.com/parlarjb/tla\\_workshop](https://github.com/parlarjb/tla_workshop)*

# Really just a first contact

- It takes an experienced person to run into details (without getting lost or mystifying others)
  - Math, math, math
  - not to speak of insights what abstractions for modeling given system will work the best (accuracy of the model vs. its brevity)

# Really just a first contact

- It takes an experienced person to run into details (without getting lost or mystifying others)
- I am barely able to understand *what*, and even less *how*, have still a lot to learn :-)

# Really just a first contact

- It takes an experienced person to run into details (without getting lost or mystifying others)
- I am barely able to understand *what*, and even less *how*, have still a lot to learn :-)
- Still, TLA+ seems approachable tool to formalize and reason about systems and algorithms, incl. *distributed*
- Hopefully, you are now motivated to experiment on your own

# Useful resources

- The TLA+ Video Course
  - <https://lamport.azurewebsites.net/video/videos.html>
- TLA+ cheatsheet
  - <https://lamport.azurewebsites.net/tla/summary-standalone.pdf>
- TLA+ chapter from Learn TLA+ book
  - <https://learntla.com/tla/>
  - otherwise heavily PlusCal oriented
- TLA+ organization at GitHub (esp. refer to Examples project)
  - <https://github.com/tlaplus>
- ...