Why Pony?

Bleeding edge technology for your concurrency woes
Sean T Allen

Author of Storm Applied
Member of the Pony core team
VP of Engineering at Sendence
Pony

- Open-source, object-oriented, actor-model, capabilities-secure, high-performance programming language
“A programming language is just another tool. It’s not about syntax. It’s not about expressiveness. It’s not about paradigms or models. It’s about managing hard problems.”

-Sylvan Clebsch
Why Pony?
Why Pony?

- Highly Concurrent
- Predictable latencies
- Data Safety
Highly Concurrent
Highly Concurrent

- Actor model
- Async messaging
- Work stealing scheduler
- Mechanical sympathy
Predictable latencies
Predictable Latencies

- No “stop the world” GC
- Per-actor heaps
- Better clustering
Data Safety
Data Safety

☐ Reference Capabilities
Why Not Pony?
Why Not Pony?

- Concurrency isn’t your problem
- Lack of API Stability
- “Batteries Not Included”
- Limited Native Tooling
Is Pony right for you?
Is Pony right for you?

Yes, if…

☐ You have a hard concurrency problem

☐ You aren’t reliant on a lot of existing libraries

☐ You are willing to write “most everything” from scratch
Takeaways...

Pony has a powerful, data-race free, concurrency-aware type system

The Pony runtime can help you solve hard concurrency problems

You might be able to use Pony in production now
Trash Day paper:

ORCA paper:
https://www.ponylang.org/media/papers/OGC.pdf
@seantallen
www.monkeysnatchbanana.com

@SendenceEng
www.sendence.com

@ponylang
www.ponylang.org
#ponylang on freenode