Building a Rack-Scale Computer with P4 at the Core

Ryan Goodfellow
Engineer @ Oxide Computer Company
Outline

Oxide product context
- What is a rack scale computer?
- Where does P4 fit in?
- Why is network programmability important?
- x4c compiler intro.
- Why a new compiler?
- Workflows with x4c: development, testing and CI.

Live demos!
- Emulating an ASIC in a hypervisor.
- Writing networking tools in P4.
Hardware and software designed together

Vertically integrated and scale-ready. Bringing hyperscaler agility to the mainstream enterprise.
Very custom hardware
Why is network programmability important?

**Flexibility**
- Example: Cooperative distributed NAT.

**Fit-for-purpose** resource usage.
- Using precious TCAM for exactly what our requirements call for.

**Comprehensibility**
- Our customer operators have a responsibility to their organizations to keep mission-critical networks running.
- Giving them data plane code access gives them the autonomy comprehend network behaviors and solve problems independently.
Why a new compiler?

We needed a way to run P4 code in very different contexts.

In the kernel, in userspace, embedded in Rust programs, free standing, in hypervisors, etc.

Compiling from P4 to Rust and then from Rust to whatever form we need makes a lot of sense.
Workflows with x4c

Compilation to Pipeline Library

Direct Inclusion in Rust Code
ASIC emulation in the Hypervisor
Writing networking tools in P4 and Rust
Thank you!