P4 Workshop
Welcome

Nate Foster
Cornell University

State of P4

"Our whole networking industry stands to benefit from a language like P4 that unambiguously specifies forwarding behavior, with dividends paid in software developer productivity, hardware interoperability, and furthering of open systems and customer choice."

— Tom Edsall, Cisco

State of P4

"Our whole networking industry stands to benefit from a language like P4 that unambiguously specifies forwarding behavior, with dividends paid in software developer productivity, hardware interoperability, and furthering of open systems and customer choice."

— Tom Edsall, Cisco

Industry Momentum

- Diverse collection of P4-enabled targets
- Growing number of P4-based products
- Real-world deployments

State of P4

"Our whole networking industry stands to benefit from a language like P4 that unambiguously specifies forwarding behavior, with dividends paid in software developer productivity, hardware interoperability, and furthering of open systems and customer choice."

— Tom Edsall, Cisco

Industry Momentum

- Diverse collection of P4-enabled targets
- Growing number of P4-based products
- Real-world deployments

Academic Interest

- Research papers at top conferences
- New courses at leading universities

State of P4

"Our whole networking industry stands to benefit from a language like P4 that unambiguously specifies forwarding behavior, with dividends paid in software developer productivity, hardware interoperability, and furthering of open systems and customer choice."

— Tom Edsall, Cisco

Industry Momentum

- Diverse collection of P4-enabled targets
- Growing number of P4-based products
- Real-world deployments

Academic Interest

- Research papers at top conferences
- New courses at leading universities

Open Source Community

- Vibrant technical working groups
- Powerful set of P4 tools

Agenda

Overview

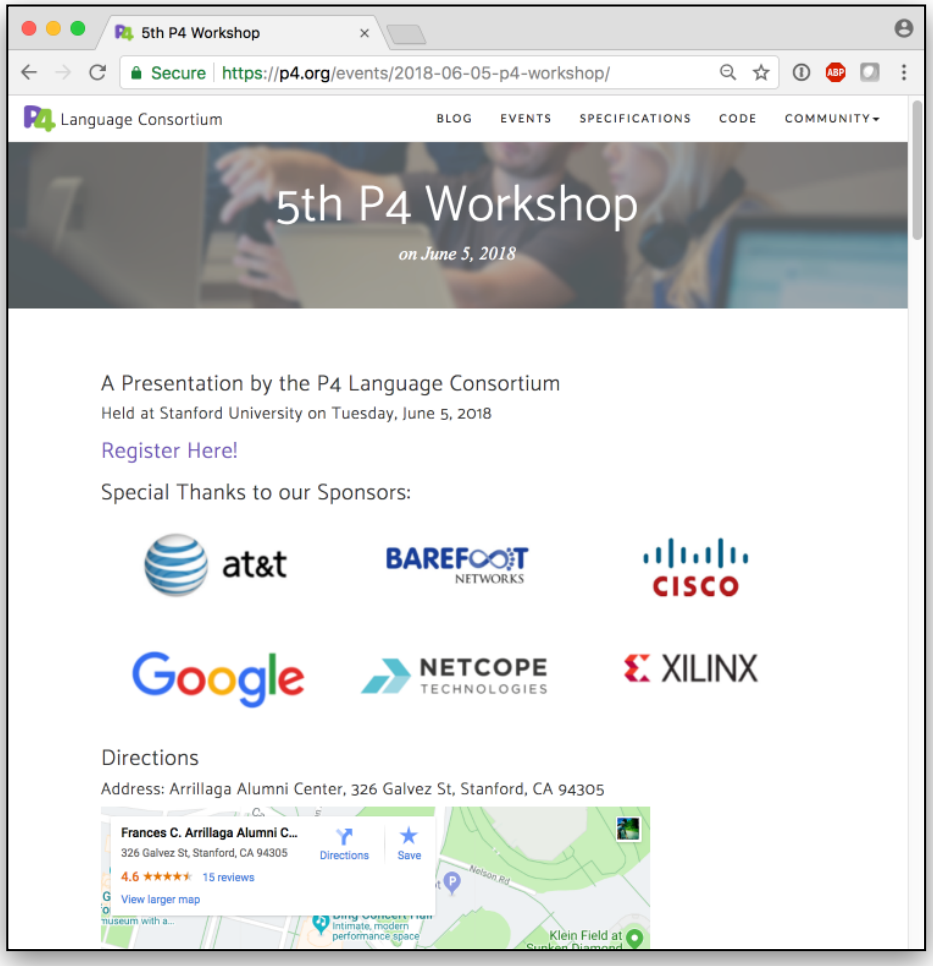
- Recent progress
- Future roadmap

Working Group Updates

- Language Design
- Architectures
- APIs
- Applications

Presentation Track

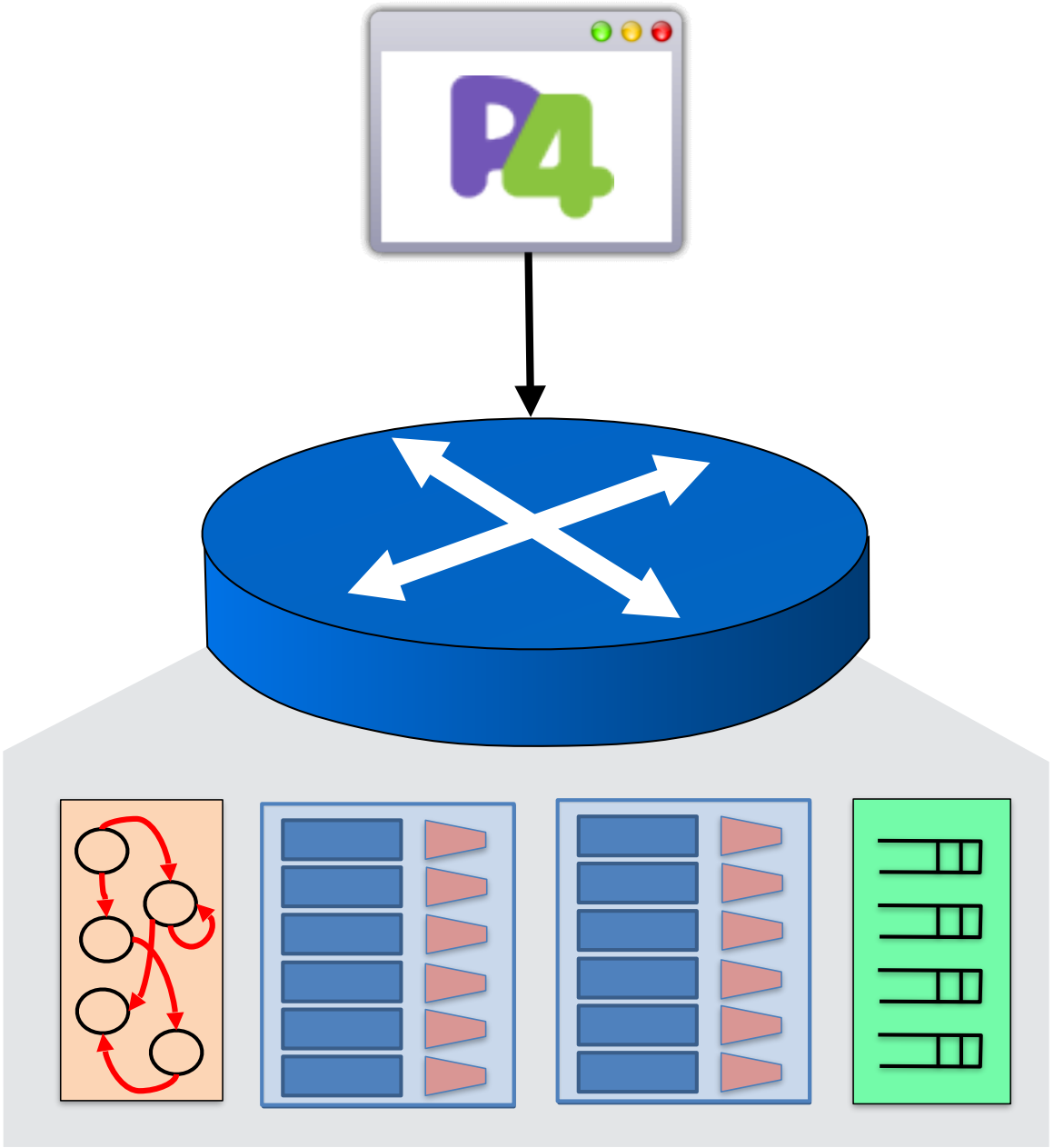
- 11 accepted talks + 1 keynote
- 20 minutes each



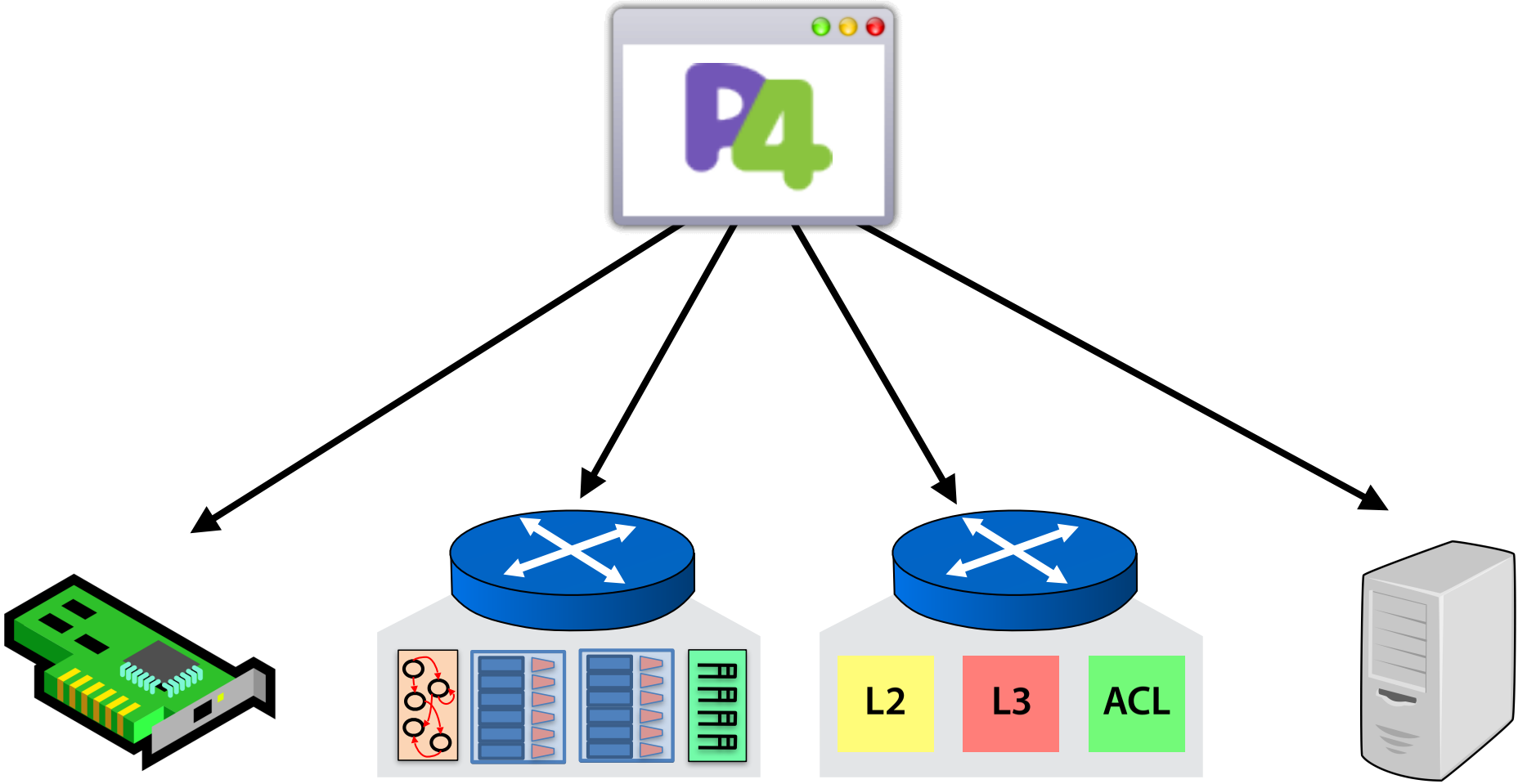
Demo Track

- 14 accepted demos
- 1 minute lightning talks + live demos(at other end of building)

P4 Perspective (ca. 2014)



Diverse Targets



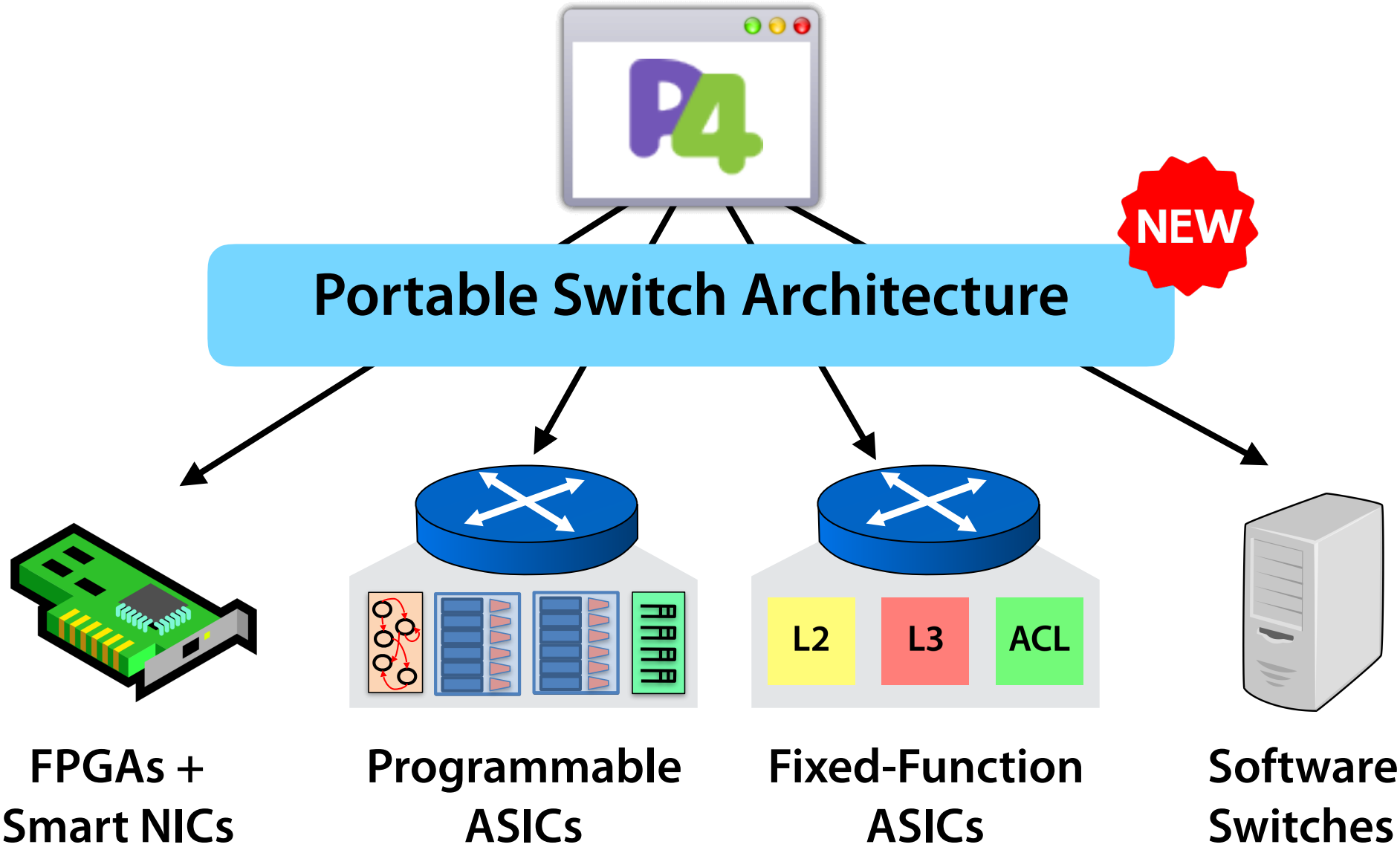
FPGAs + Smart NICs

Programmable ASICs

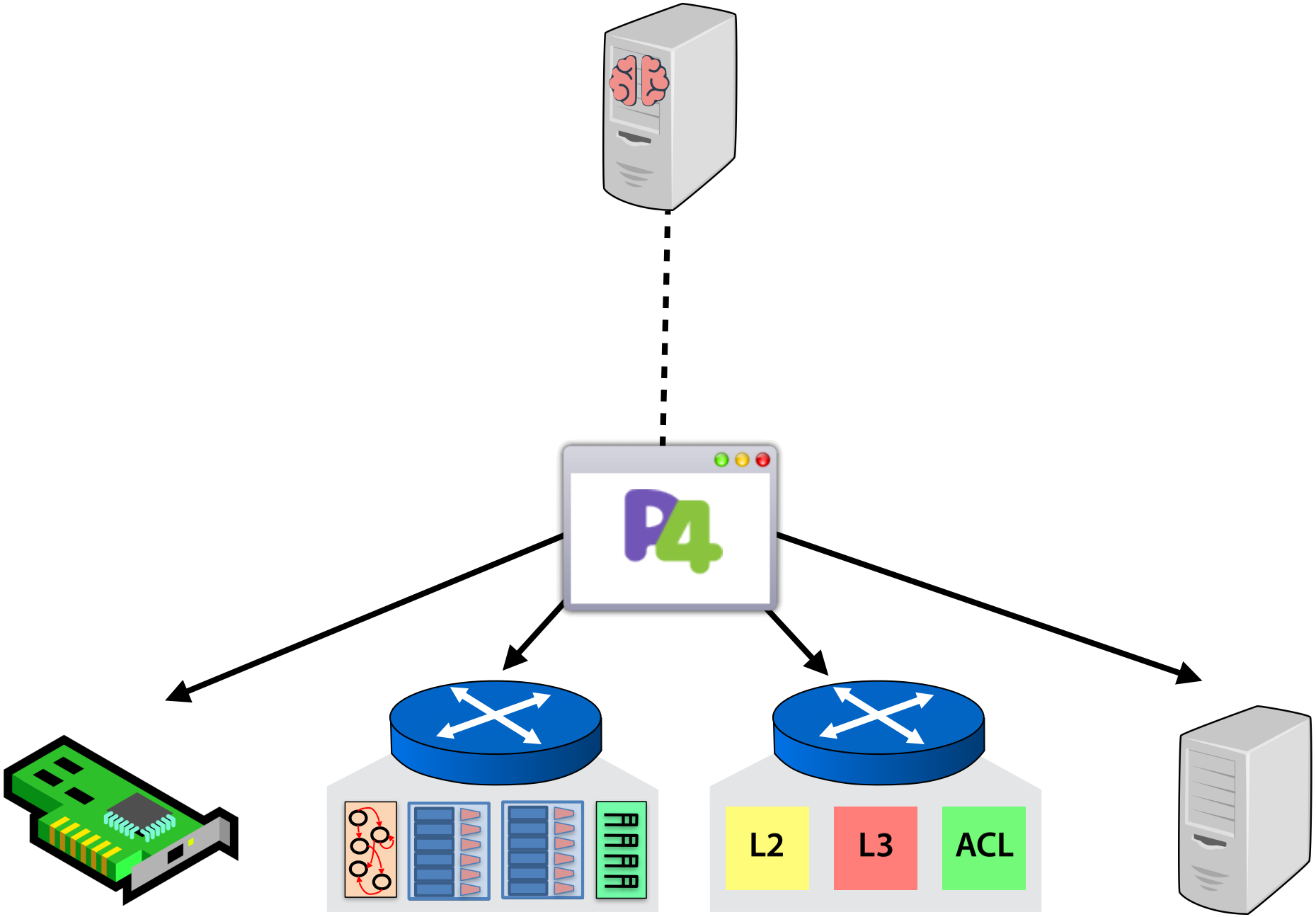
Fixed-Function ASICs

Software Switches

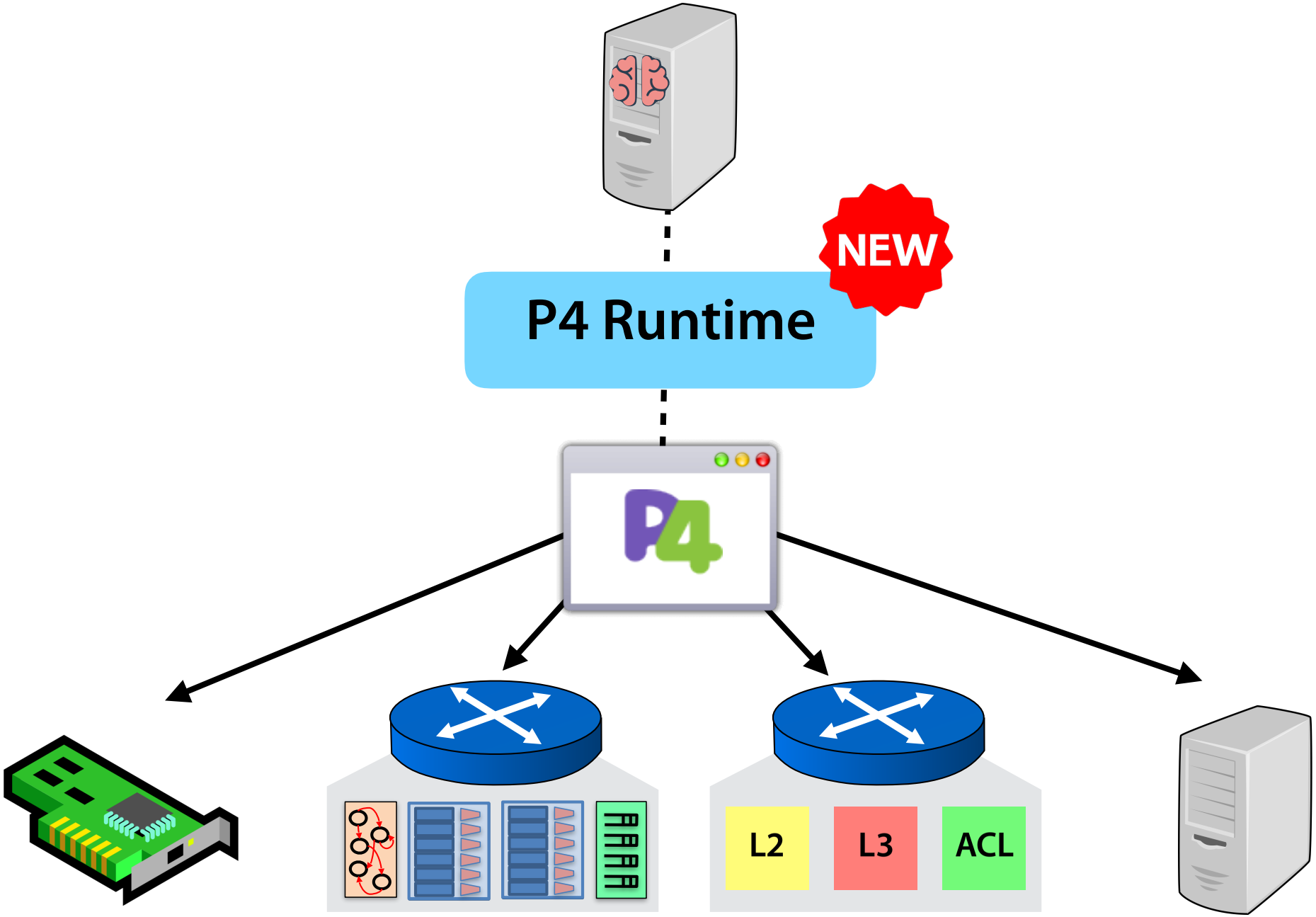
Diverse Targets



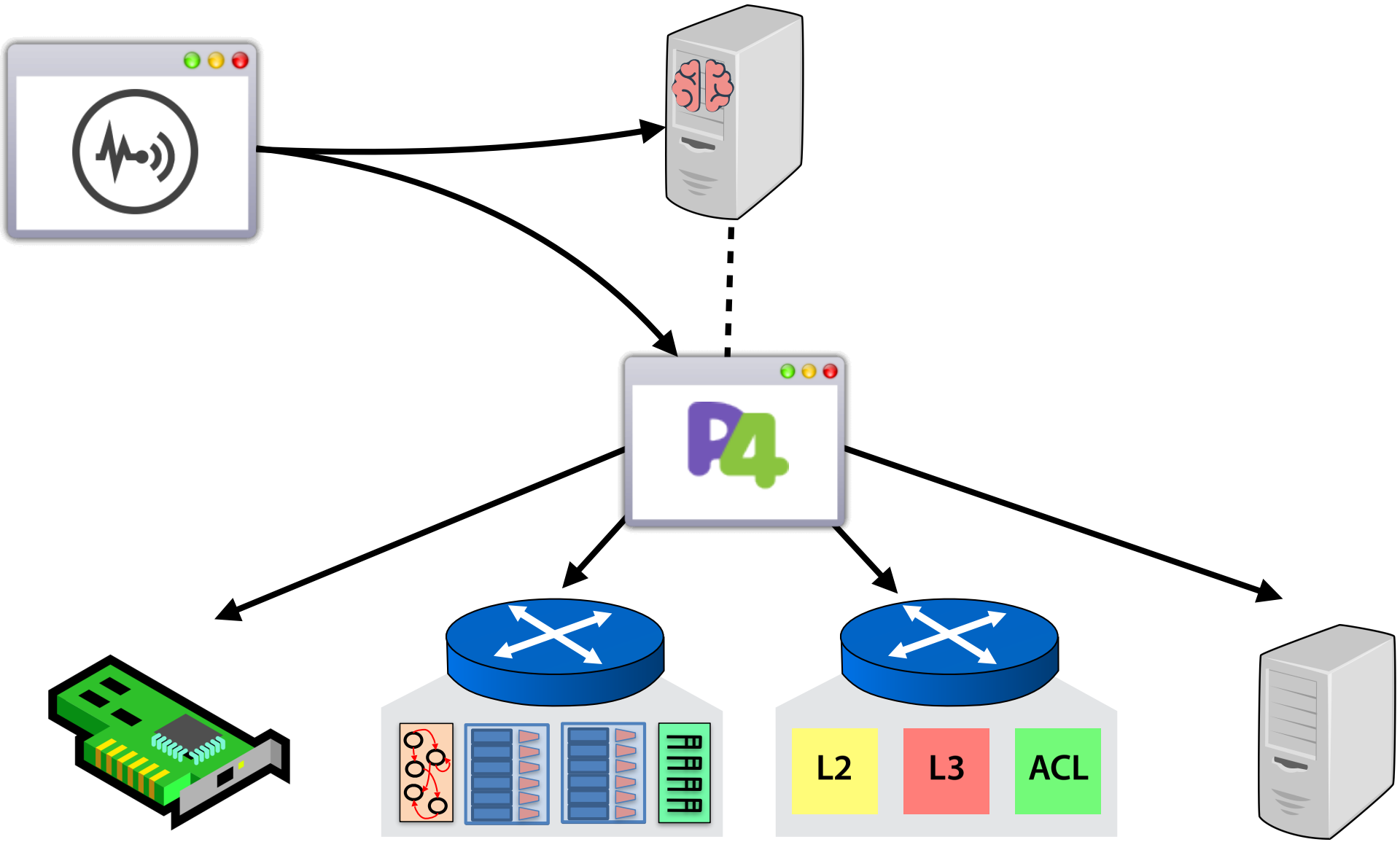
Complex Control-Planes



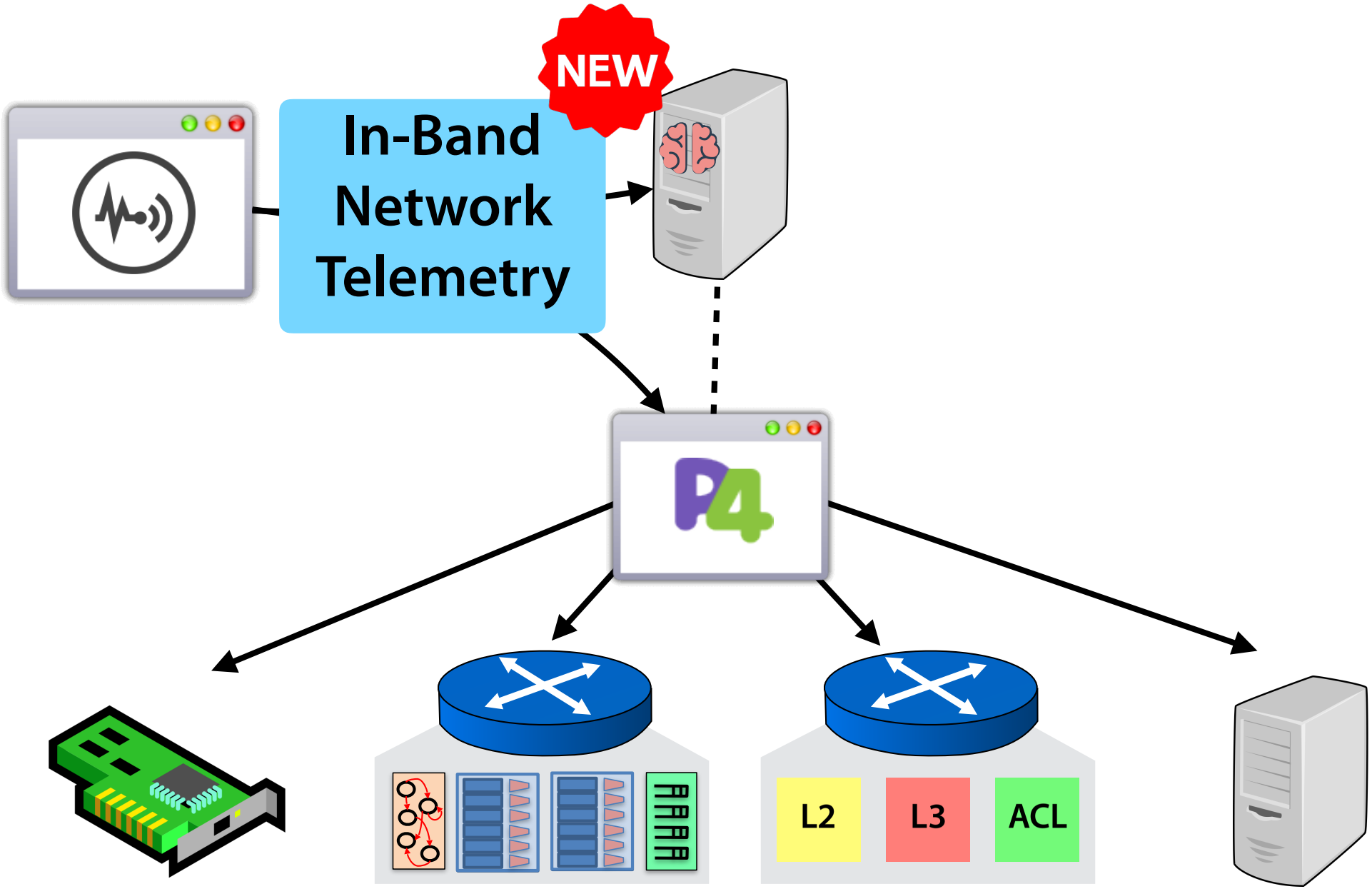
Complex Control-Planes



Rich Applications



Rich Applications



Language Design Working Group

Language Design Working Group

Last year: Released $P4_{16}$ v1.0.0

- A major update to the language:
- Target-architecture separation
- Static types
- Higher-level programming constructs

Language Design Working Group

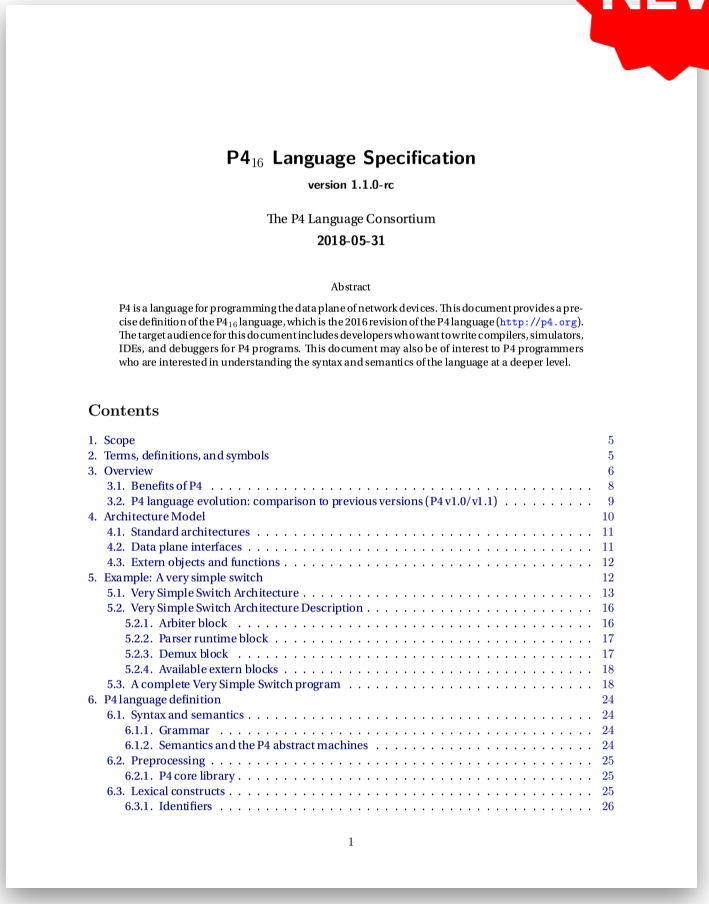
Last year: Released $P4_{16}$ v1.0.0

- A major update to the language:
- Target-architecture separation
- Static types
- Higher-level programming constructs

This year: Releasing $P4_{16}$ v1.1.0

- Backwards-compatible changes to improve usability
- Top-level functions
- Optional and named parameters
- Type definitions
- Enum representations
- Parser value sets
- Saturating arithmetic
- Structured annotations

NEW



$P4_{16}$ Language Specification

version 1.1.0-rc

The P4 Language Consortium

2018-05-31

Abstract

P4 is a language for programming the data plane of network devices. This document provides a precise definition of the $P4_{16}$ language, which is the 2016 revision of the P4 language (http://p4.org). The target audience for this document includes developers who want to write compilers, simulators, IDEs, and debuggers for P4 programs. This document may also be of interest to P4 programmers who are interested in understanding the syntax and semantics of the language at a deeper level.

Contents

1

Ergonomic Improvements

Top-Level Functions

```
bit<32> max(in bit<32> left, in bit<32> right) {
    if (left > right)
        return left;
    return right;
}
```

Named Parameters

```
extern void f(in bit<32> x, out bit<16> y);
bit<32> xa = 0;
bit<16> ya = 1;
f(xa, ya);           //by position
f(x = xa, y = ya);   //by name
f(y = ya, x = xa);   //by name
```

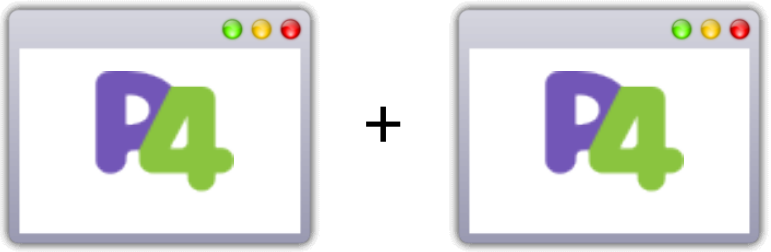
Enum Representations

```
enum bit<8> E {
  e1 = 0,
  e2 = 1,
  e3 = 2
}
```

Roadmap

Modularity

- Construct complex programs by out of simpler pieces of P4 code
- Design tools for separately compiling and linking programs
- Develop constructs for controlling names, imports, exports, etc.



Semantics

- Precisely define the packet-processing behavior of every program
- Build reference implementations that can be used for testing and validation
- Use semantics to guide design of future language extensions

LDWG Co-Chairs

Gordon Brebner
Xilinx

Mihai Budiu
VMware

P4 Distinguished Service Award

Gordon Brebner
Xilinx

Citation: *For dedicated service to the P4 community as co-chair of the Language Design Working Group.*

As one of the stewards of the P4 language, Gordon has provided essential leadership and guidance through the early years of the development of the language. He has been a tireless advocate for the elegant and yet pragmatic design that can be seen in many of the language's features. At Xilinx, Gordon also led development of a P4-enabled target, building on his decades of expertise implementing efficient packet-processing engines using FPGAs.

Education Working Group

Charter: The purpose of this working group is to promote the education of P4 programmers. Its main tasks will include the hosting and curating of shared community resources; the development of instructional materials; and the organization of educational events.

Co-Chairs

Robert Soule
Lugano

Noa Zilberman
Cambridge

P4 EU Workshop @ ICNP '18
P4 Tutorial @ SIGCOMM '18


P4
.org

+


THE
LINUX
FOUNDATION


Open Networking Foundation

Get Involved

Become a member of P4.org

- No fee + simple membership agreement
- Code and data under Apache2 License

Participate in working groups

- Activities are open to all members
- Anyone with a good idea can help shape the future of P4

Contribute to Open-Source Software

- Compiler (p4c)
- Software switch (bmv2)
- Control-plane APIs (P4Runtime)
- Tutorials
- Documentation
- Standard applications (PSA, INT)
- New applications

Thank You

Program Committee

- Andy Fingerhut, Cisco (chair)
- Mina Arashloo, Princeton
- Sujata Banerjee, VMware
- Tom Rodeheffer, Google
- Cole Schlesinger, Barefoot
- Anirudh Sivaraman, NYU
- Noa Zilberman, Cambridge

Conference Organization

- Sedef Ozcana, P4.org
- Rachel Everman, Barefoot
- Prem Jonnalagadda, Barefoot

P4.org Board

- Jennifer Rexford, Princeton
- Amin Vadhat, Google
- Nick McKeown, Stanford

Industrial Sponsors

at&t

BAREFOOT NETWORKS

CISCO

Google

NETCOPE TECHNOLOGIES

XILINX