LABS.NETWORKRELIABILITY.ENGINEERING

Learning Automation Without Barriers Using Antidote and NRE Labs

Matt Oswalt - @Mierdin

Juniper Networks

About Me

Matt Oswalt
@mierdin

- ✔ 8 years in the industry
  - Network engineer
  - Consultant
  - Developer
  - Open Source Maintainer
- ✔ Currently at Juniper Networks
- ✔ Passionate about skills evolution

NRE LABS

Network Reliability Engineering

Nre

Core networking fundamentals still matter. It's right in the name.

n**R**e

Represents a better way of doing things. Emphasizes the true goal of automation

nr**E**

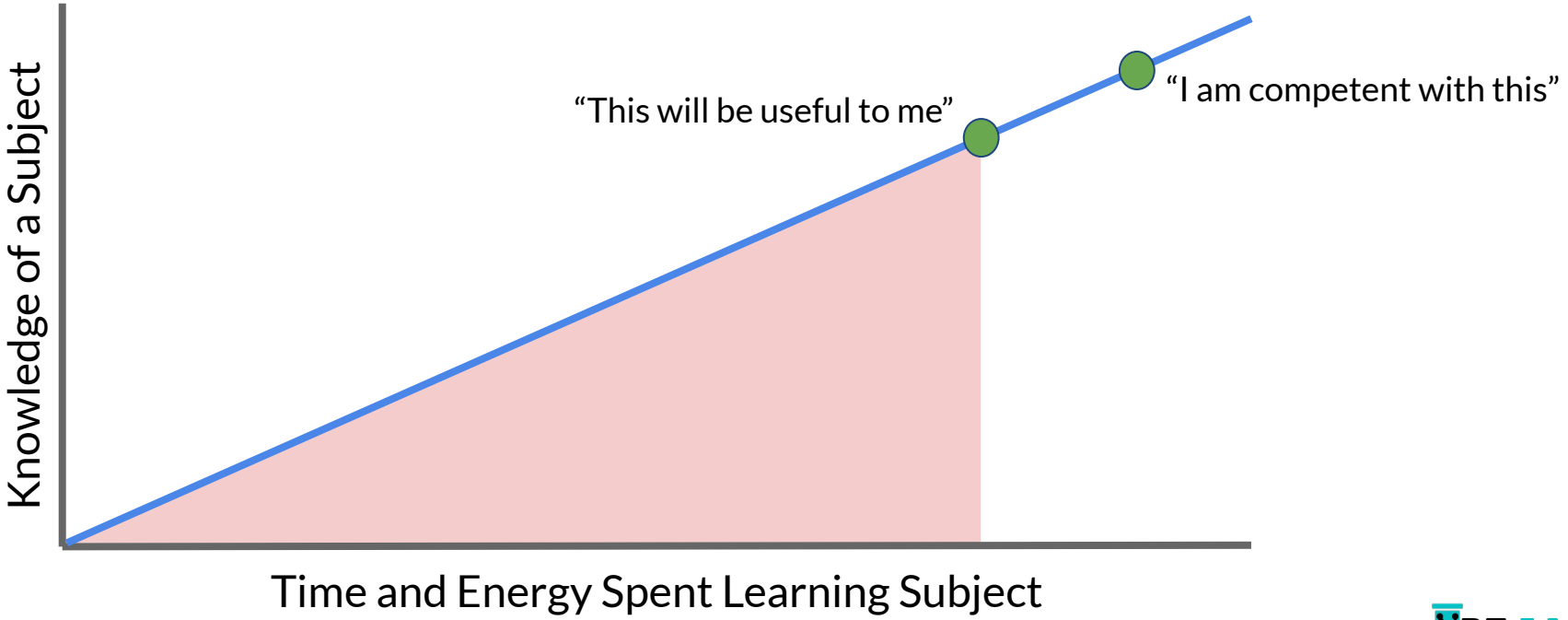
You can't buy engineering - you DO it. Sidesteps the "productization" of automation

Codify Automate Test

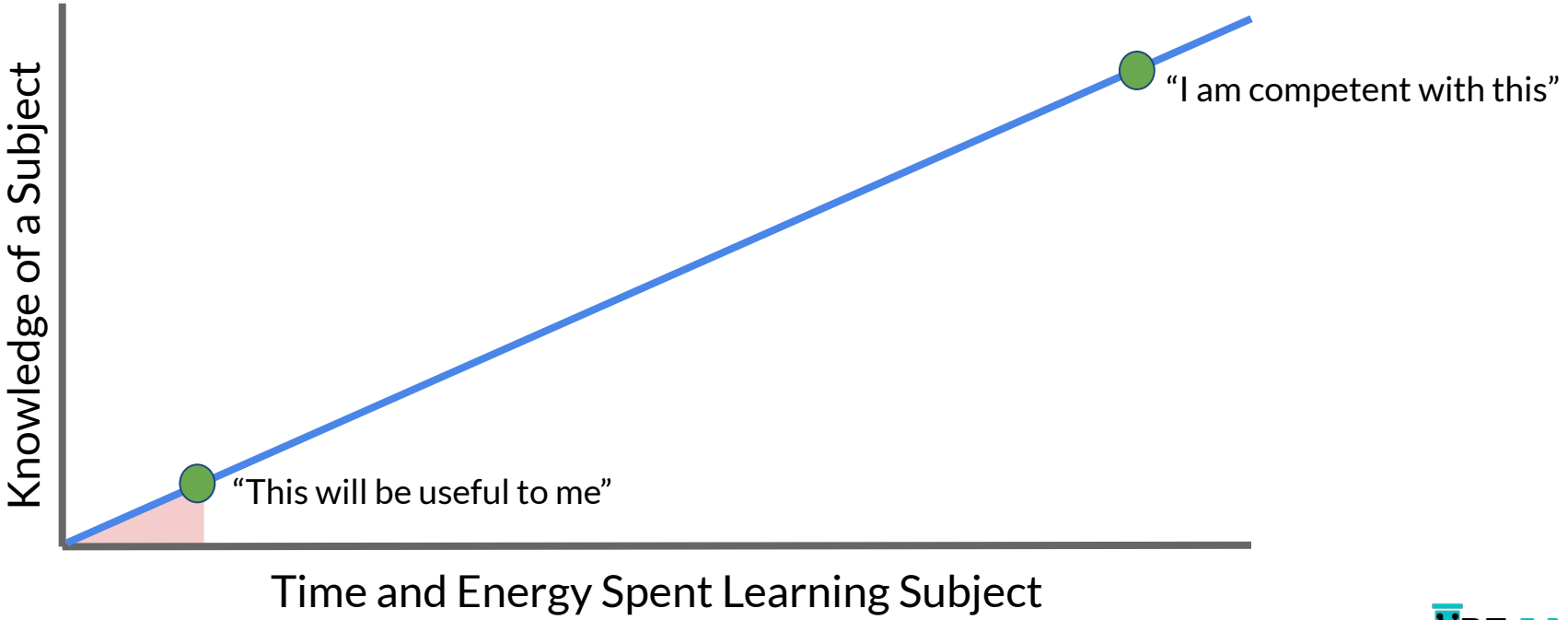
Monitor Measure

NRE LABS

Time Investment Minimum (TIM) - HIGH



Time Investment Minimum (TIM) - LOW



NRE LABS

NRE Labs

Community platform for learning and teaching automation and Network Reliability Engineering

- Totally browser-based
- Free - no login, paywall or creepy trackers
- Vendor-neutral
- Open Source (curriculum too!)

DEMO

NRE LABS

```
cd /antidote/lessons/lesson-12/
cat jsnapy_tests.yaml
```

Run this snippet

To review, these tests assert:

- There must be one BGP group configured
- There must be two BGP peers configured
- There must not be any "down" BGP peers

In this part (Part 2), our routers have been configured with the correct BGP peers. We can verify this by checking on the current BGP summary:

```
show bgp summary
```

Run this snippet

It *looks* good, but as they say, "successful tests or it didn't happen". Let's re-run JSNAPy to make sure our tests are passing with the new configuration:

```
jsnapy --snapcheck -f jsnapy_config.yaml -v
```

Run this snippet

This time, our network is behaving the way we've declared in the tests, so they pass. It's important to note that our tests not only assert that the right configuration exists, but that the operational state of each router's BGP peer status is correct. This is a nice feature of JSNAPy - it can make assertions over anything in the entire Junos data model.

This was a lightning-quick introduction to JSNAPy. Please see the wiki for more details - there's a lot more capability than we covered here.

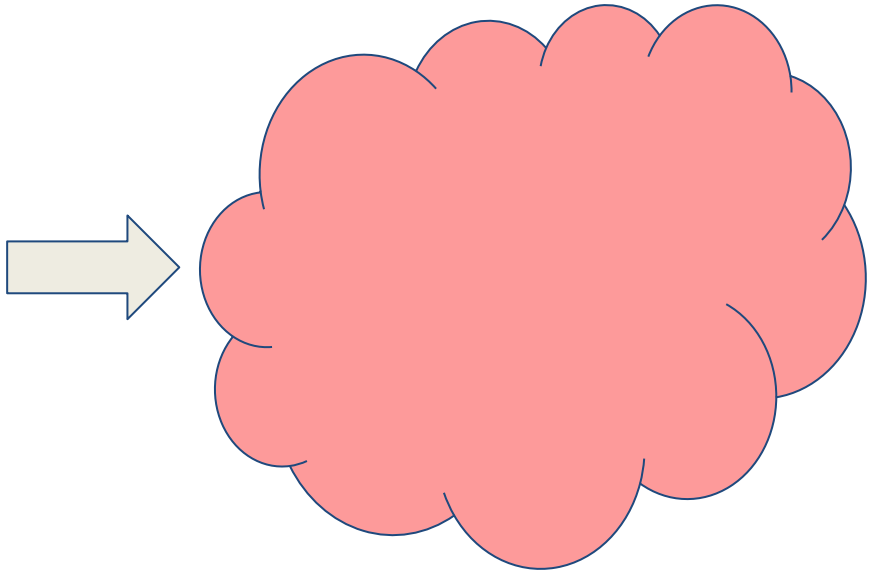
That's it for this lesson!

Use the tabs below to use this lesson's resources. Play around and explore, they're yours!

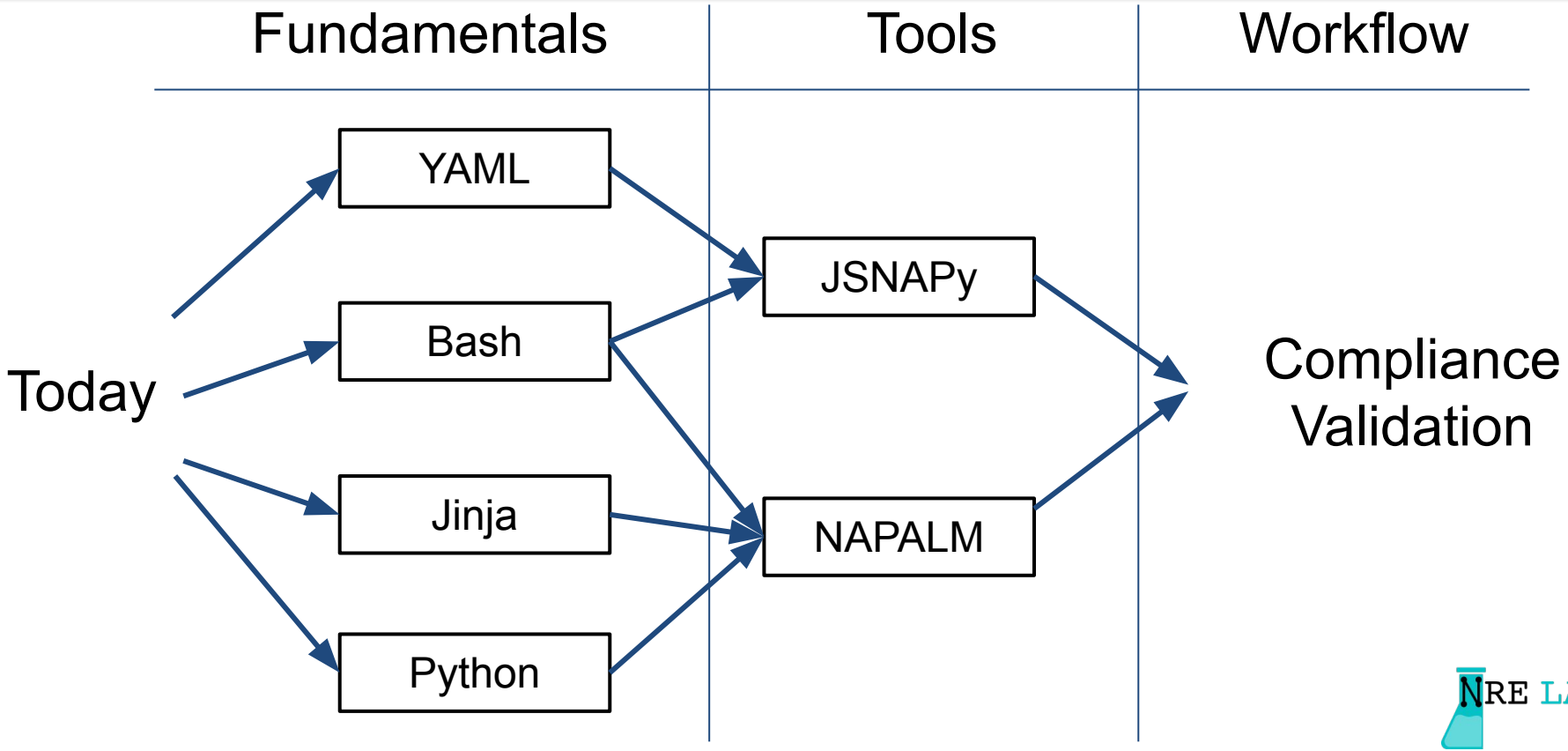
linux1 | vqfx1 | vqfx2 | vqfx3

```
**********RPC is get-bgp-summary-information**********
--------------------Performing is-equal Test Operation---------------------
Test succeeded!! BGP group count is: <1>
PASS | All "group-count" is equal to "1" [ 1 matched ]
--------------------Performing is-equal Test Operation---------------------
Test succeeded!! BGP group configured peer count is: <2>
PASS | All "peer-count" is equal to "2" [ 1 matched ]
--------------------Performing is-equal Test Operation---------------------
Test succeeded!! BGP down peer count is: <0>
PASS | All "down-peer-count" is equal to "0" [ 1 matched ]
------------------------------ Final Result!! ------------------------------
test_rpc_bgp : Passed
Total No of tests passed: 3
Total No of tests failed: 0
Overall Tests passed!!!
Connecting to device vqfx1 ..............
Tests Included : test_rpc_bgp
Taking snapshot of RPC: get-bgp-summary-information
***************************** Device: vqfx1 ******************************
Tests Included: test_rpc_bgp
**********RPC is get-bgp-summary-information**********
--------------------Performing is-equal Test Operation---------------------
Test succeeded!! BGP group count is: <1>
PASS | All "group-count" is equal to "1" [ 1 matched ]
--------------------Performing is-equal Test Operation---------------------
Test succeeded!! BGP group configured peer count is: <2>
PASS | All "peer-count" is equal to "2" [ 1 matched ]
--------------------Performing is-equal Test Operation---------------------
Test succeeded!! BGP down peer count is: <0>
PASS | All "down-peer-count" is equal to "0" [ 1 matched ]
------------------------------ Final Result!! ------------------------------
test_rpc_bgp : Passed
Total No of tests passed: 3
Total No of tests failed: 0
Overall Tests passed!!!
antidote@linux1:/antidote$
```

Today



Automated Workflow


Fundamentals
Tools
Workflow
YAML
Bash
Jinja
Python
JSNAPy
NAPALM
Today
Compliance
Validation

Why should I care?

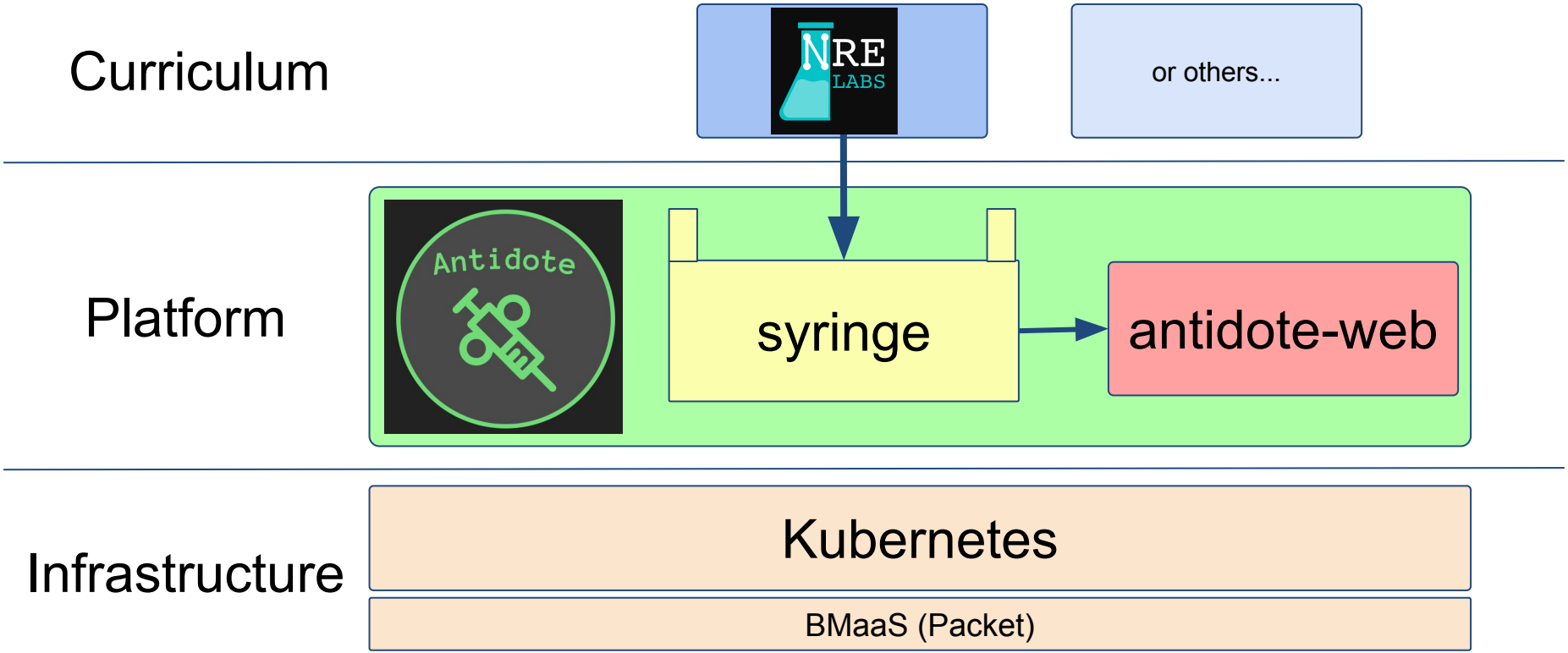
- ✔ Chance for the community to take back control of ops education
- ✔ Fairly new project - lots to do
- ✔ Covers a wide spectrum of disciplines
  - frontend, systems programming, ops, content
- ✔ Several cutting-edge technologies in use now or in the near future

NRE LABS

Creating Lessons

NRE LABS

Antidote Architectural Overview



nre-learning / **nrelabs-curriculum**

Unwatch 19 | Unstar 89 | Fork 44

Code | Issues 47 | Pull requests 7 | Projects 1 | Security | Insights | Settings

Learn network automation, all in your browser. https://labs.networkreliability.engin…

Edit

Manage topics

869 commits | **18** branches | **14** releases | **16** contributors | Apache-2.0

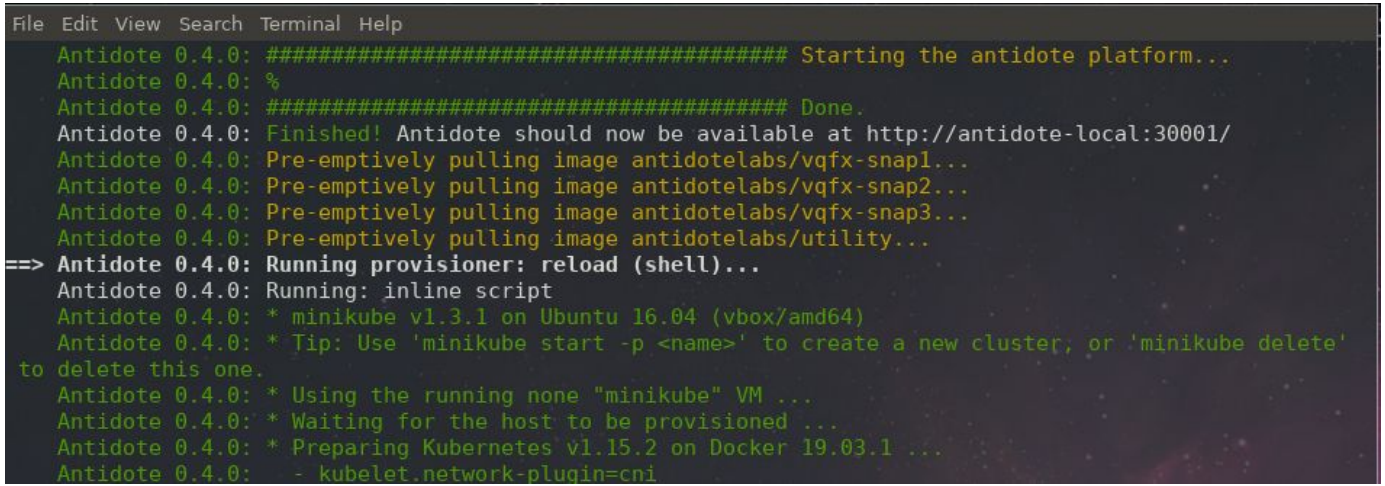
Branch: **master** | New pull request | Create new file | Upload files | Find File | Clone or download

Mierdin Fix problems with lesson 32 stage 3 ··· Latest commit `fa0cf3d` 8 days ago

collections	Changes to twin bridges collection based on feedback from kirk	14 days ago
images	Install jsnapy manually to get the yaml fix	11 days ago
lessons	Fix problems with lesson 32 stage 3	8 days ago
.dockerignore	A bit of catchup	last year
.gitignore	Add images to gitignore	5 months ago
.travis.yml	Clean up old validation logic	3 months ago
CHANGELOG.md	updated changelog	11 days ago

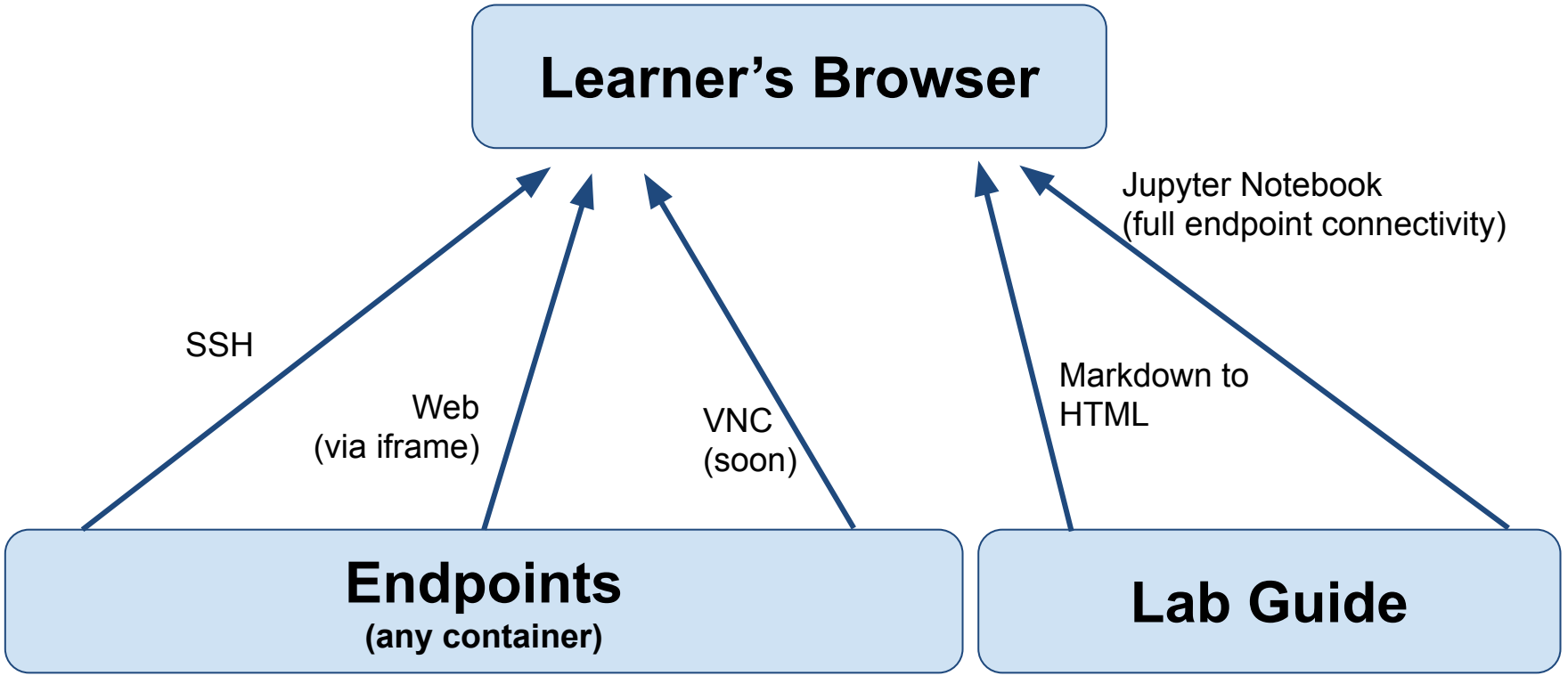
Selfmedicate

- ✔ Scripts and Kubernetes Manifests for deploying Antidote on Minikube
- ✔ https://github.com/nre-learning/antidote-selfmedicate/
- ✔ Easily preview curriculum content locally before submitting a PR.



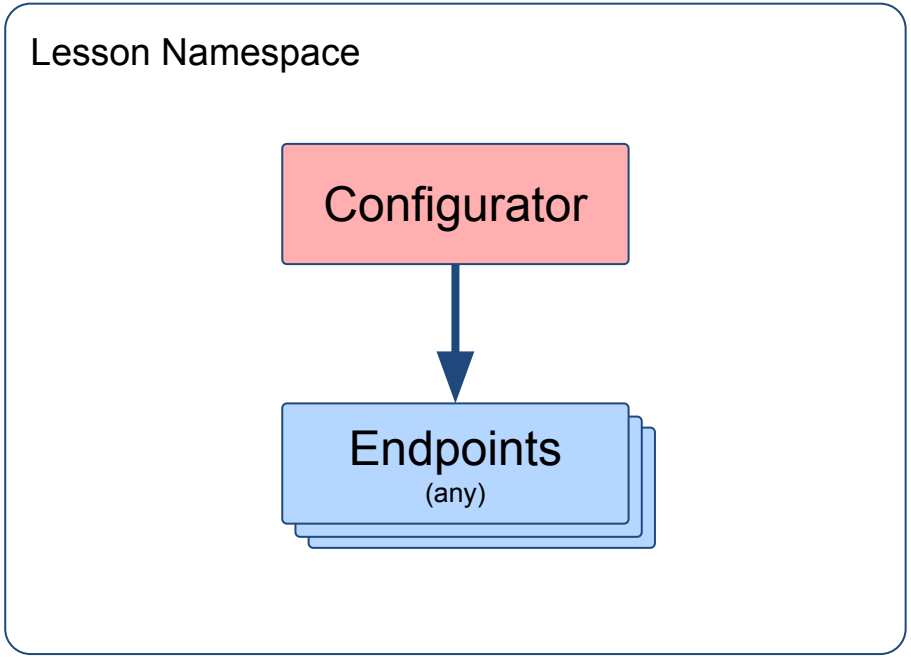
NRE LABS

Flexible Presentation Layer



Inter-Stage Configuration

- ✔ Hands-Free environment prep while moving within a lesson
- ✔ Configurations present within lesson directory will be applied during stage transitions
- ✔ Currently supports NAPALM, Ansible, or Custom Python



NRE LABS

Network Devices in Docker

- Originally inspired by vrnetlab but currently a bit more bespoke. Hoping to get more standardized soon.
- Images packaged straight into docker and executed by the kubelet on the scheduled host
- ANY vendor is feasible as long as it runs in a VM and talks on a port

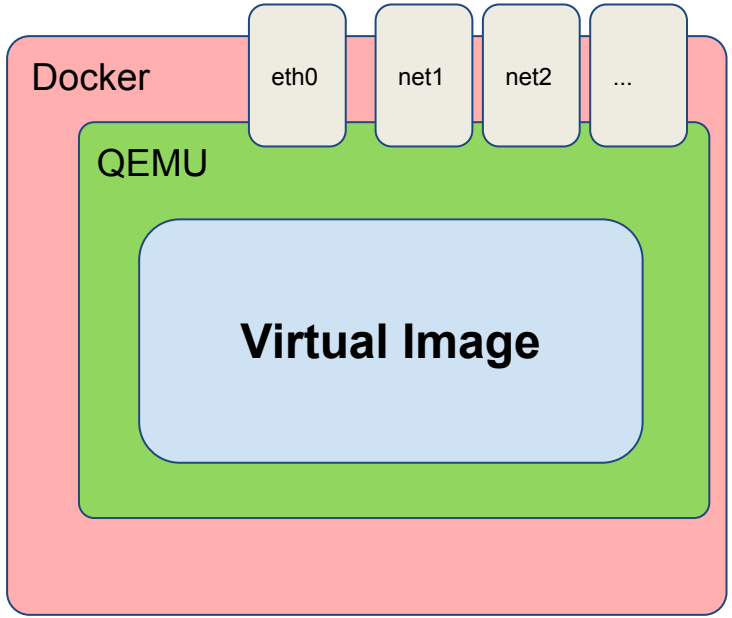
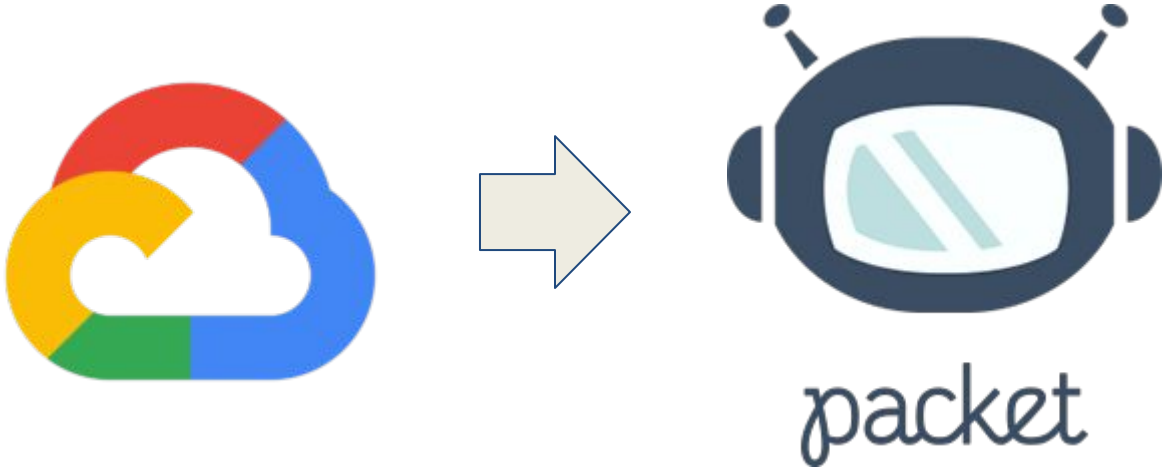


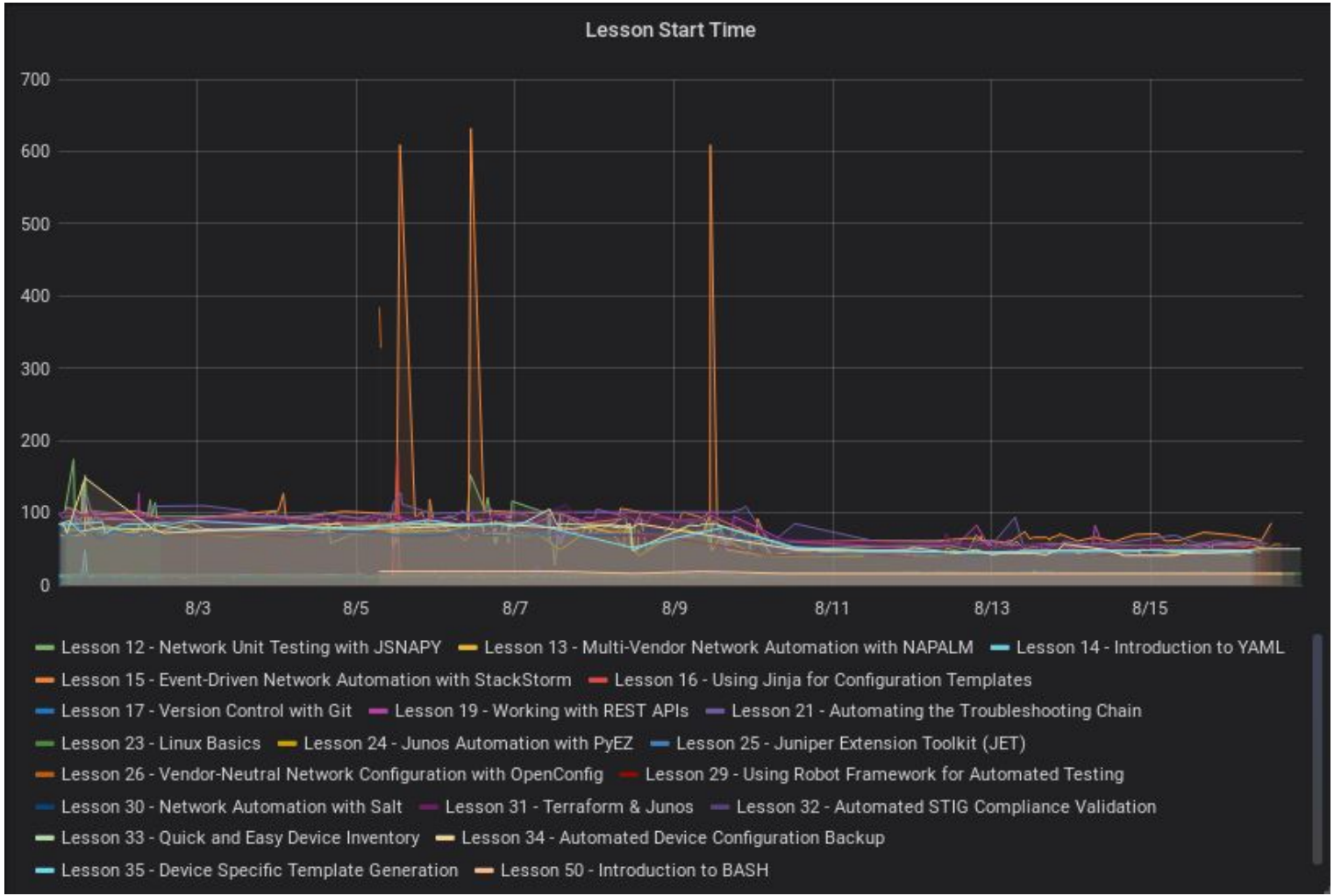
Image Catalog

Current
- vQFX
- vMX
- Cumulus VX

Soon
- VyOS
- ExtremeXOS

NRE LABS

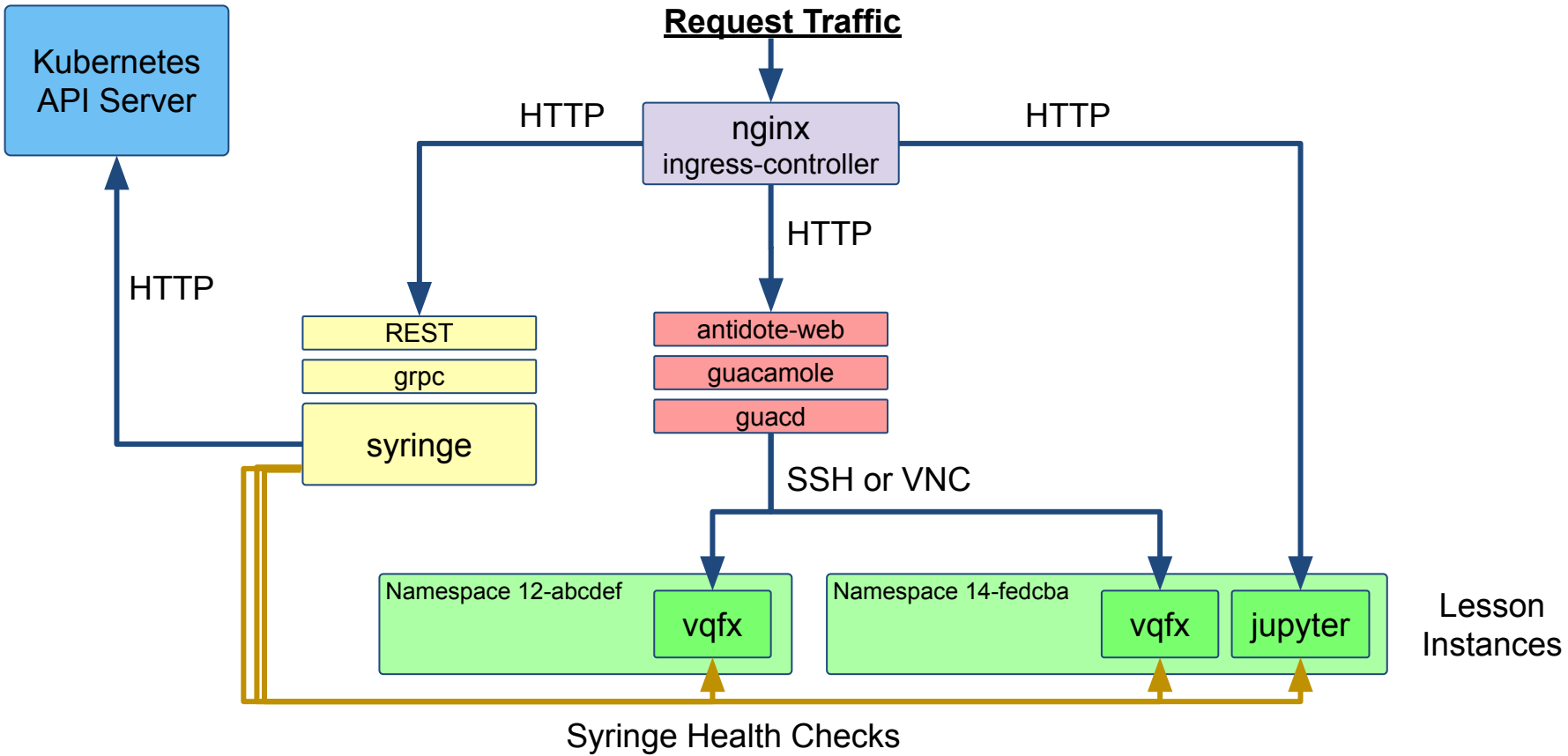



Lesson Start Time
700
600
500
400
300
200
100
0
8/3
8/5
8/7
8/9
8/11
8/13
8/15
Lesson 12 - Network Unit Testing with JSNAPY
Lesson 13 - Multi-Vendor Network Automation with NAPALM
Lesson 14 - Introduction to YAML
Lesson 15 - Event-Driven Network Automation with StackStorm
Lesson 16 - Using Jinja for Configuration Templates
Lesson 17 - Version Control with Git
Lesson 19 - Working with REST APIs
Lesson 21 - Automating the Troubleshooting Chain
Lesson 23 - Linux Basics
Lesson 24 - Junos Automation with PyEZ
Lesson 25 - Juniper Extension Toolkit (JET)
Lesson 26 - Vendor-Neutral Network Configuration with OpenConfig
Lesson 29 - Using Robot Framework for Automated Testing
Lesson 30 - Network Automation with Salt
Lesson 31 - Terraform & Junos
Lesson 32 - Automated STIG Compliance Validation
Lesson 33 - Quick and Easy Device Inventory
Lesson 34 - Automated Device Configuration Backup
Lesson 35 - Device Specific Template Generation
Lesson 50 - Introduction to BASH

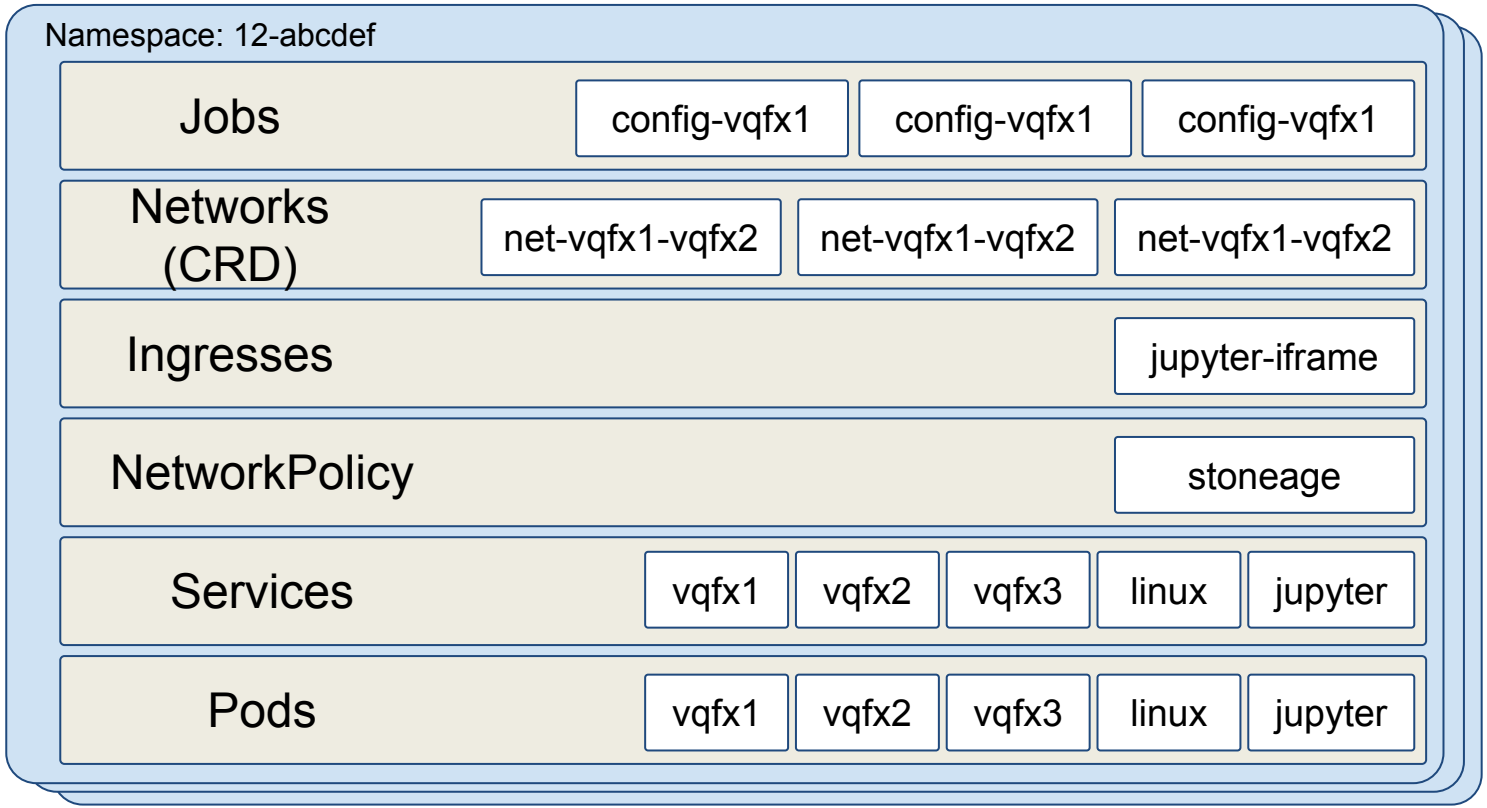
Diving Deeper...

NRE LABS

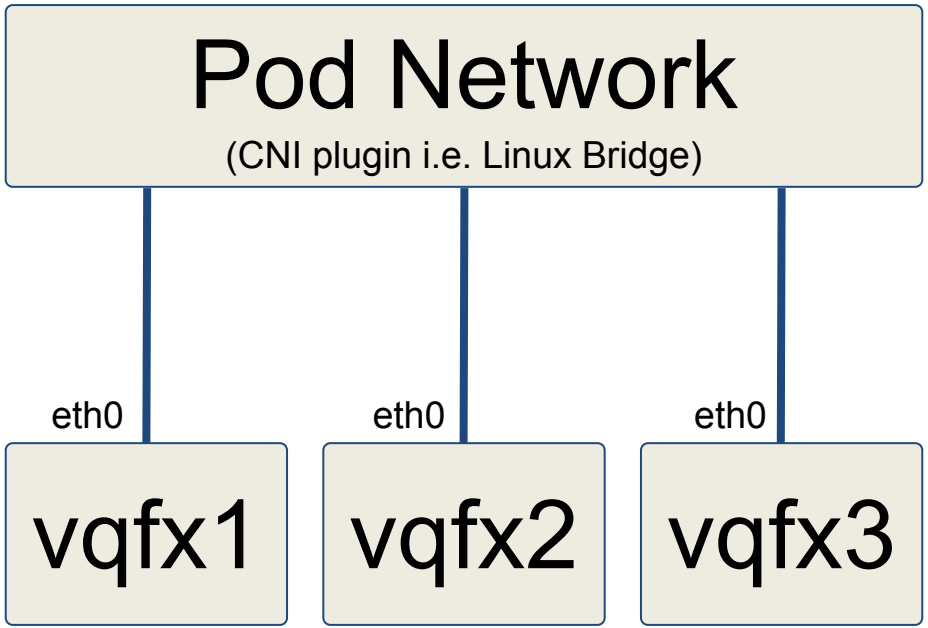
Antidote as Deployed in Kubernetes



Anatomy of a Lesson



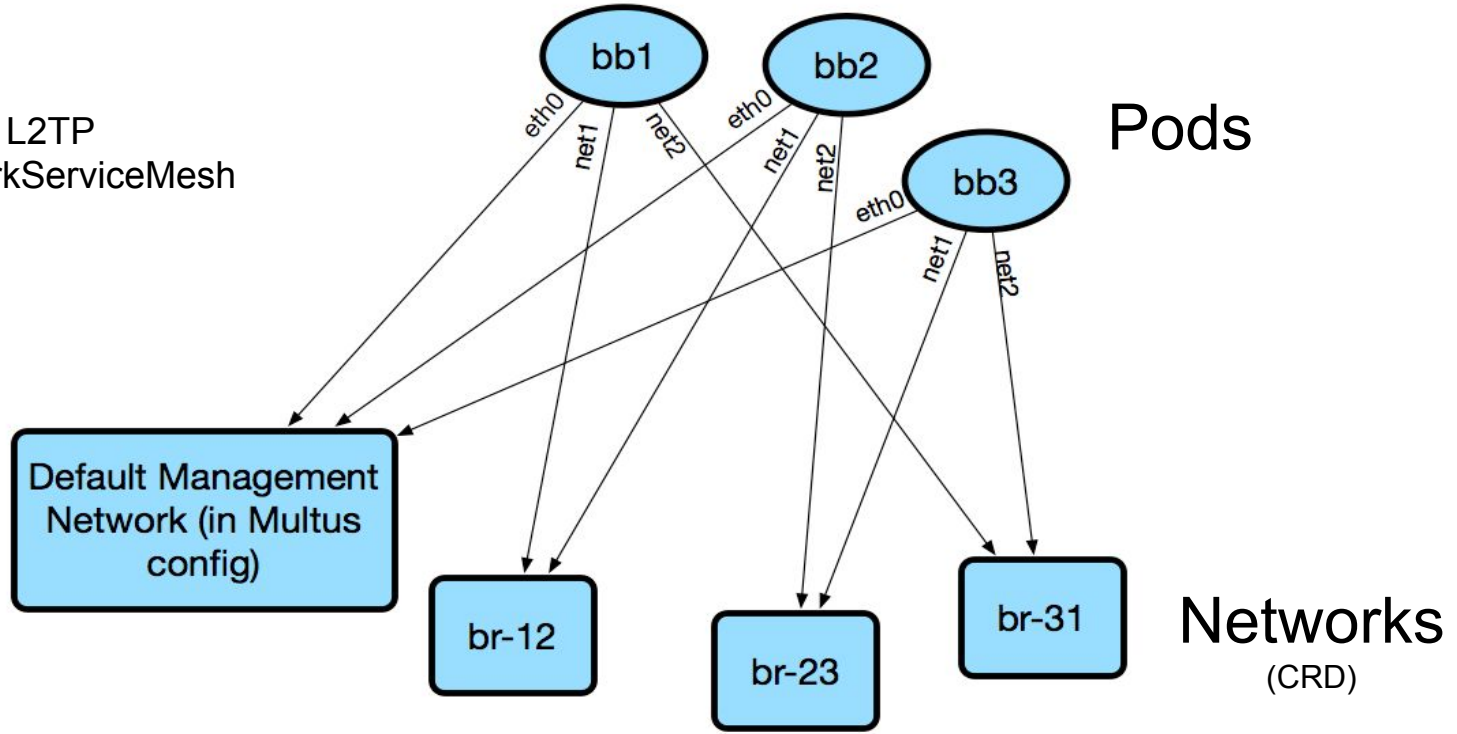
Normal Kubernetes Pod Networking



Using Multus for Advanced Network Topologies

Alternatives:

- QEMU L2TP
- NetworkServiceMesh

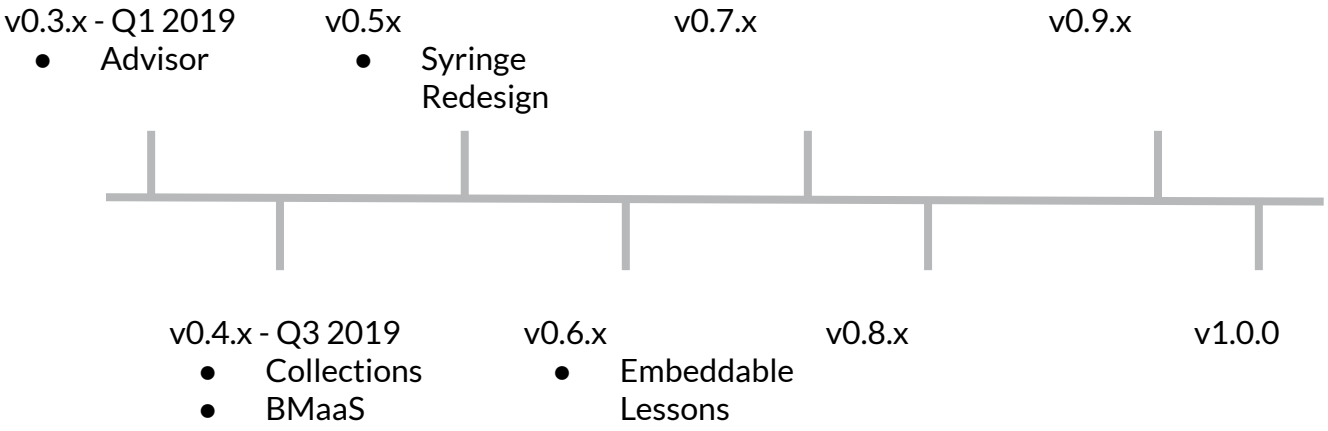


https://github.com/intel/multus-cni

The Road to v1.0

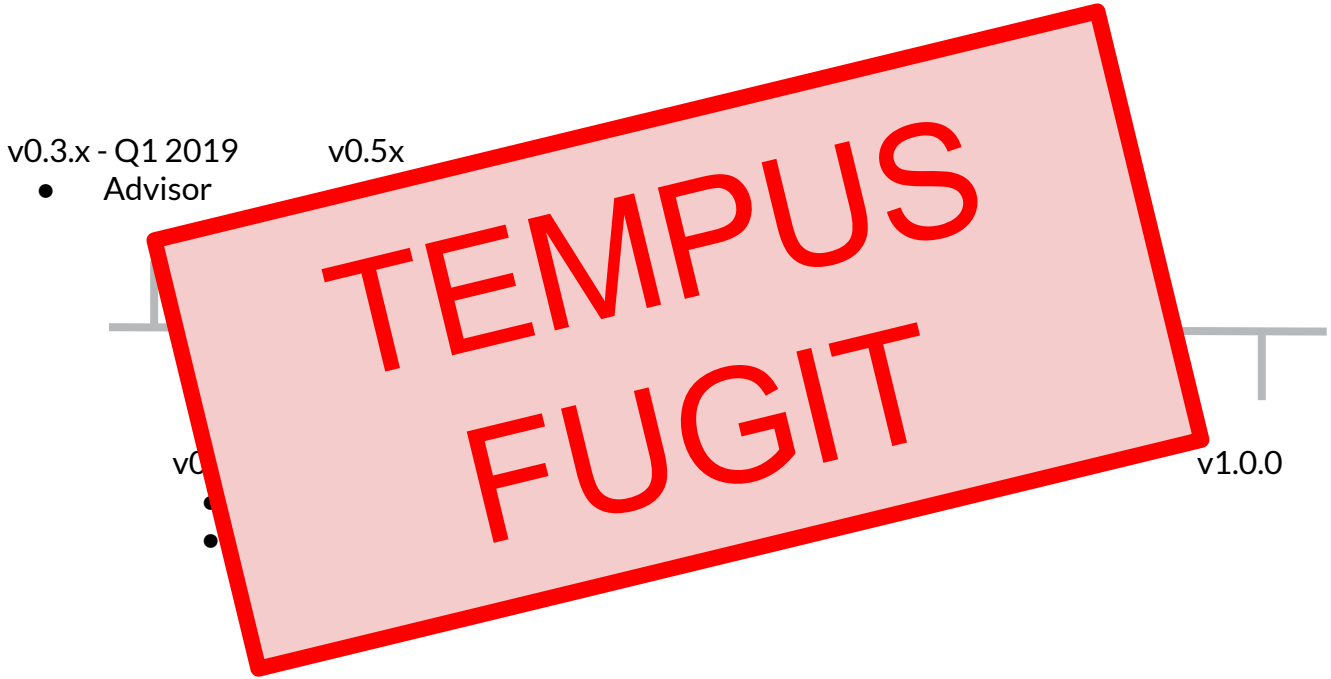
NRE LABS

The Road to v1.0



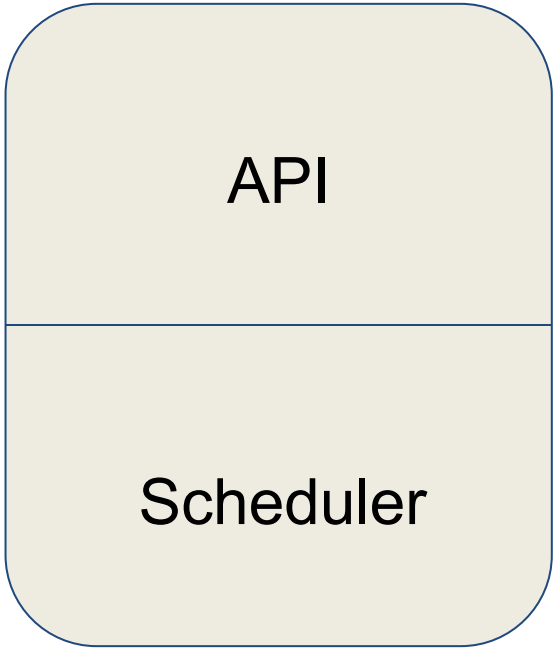
NRE LABS

The Road to v1.0



NRE LABS

Current Syringe Architecture



Single Process - “syringed”

Advantages:

- Single binary
- No external database to worry about
- Allowed us to get NRE Labs public quick

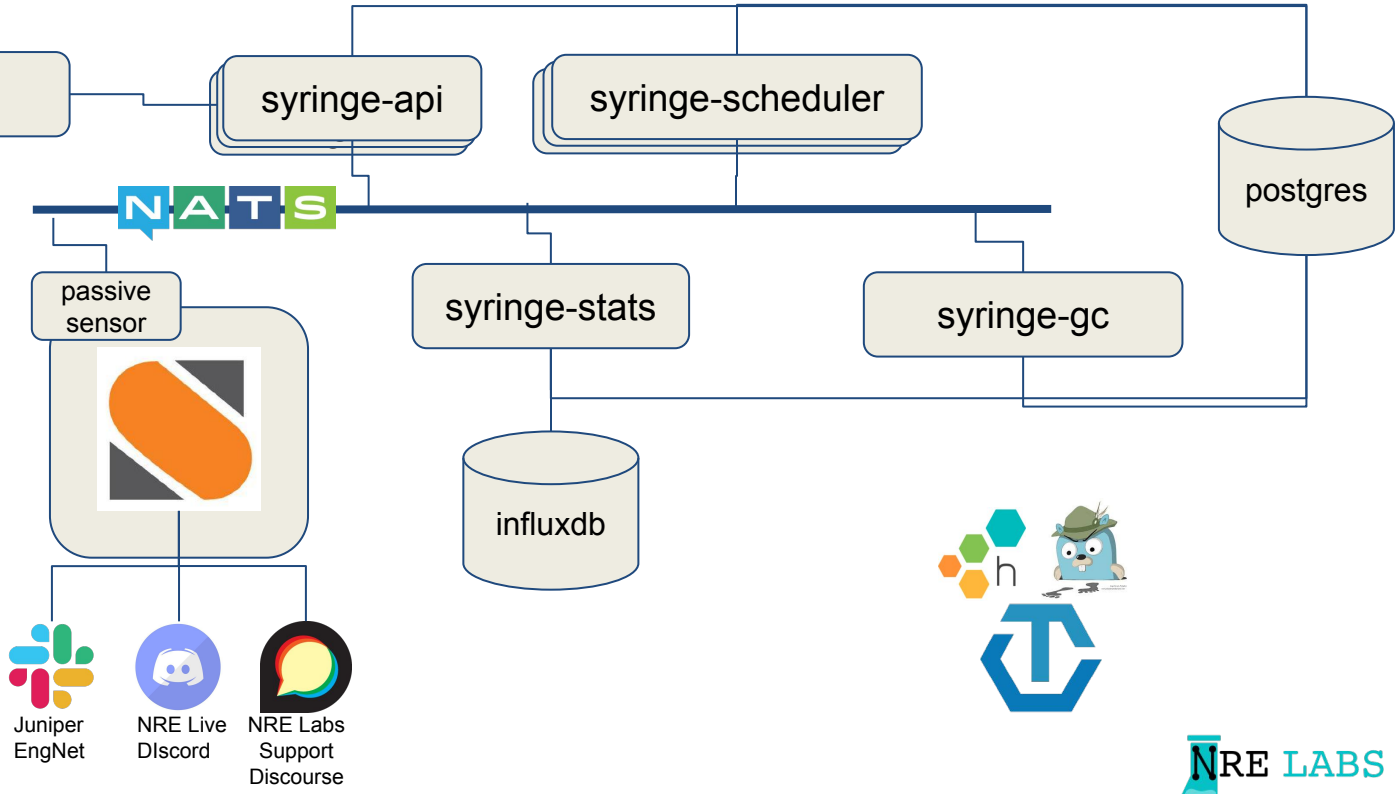
Disadvantages

- Single point of failure
- Everything is tightly coupled, harder to extend
- State is kept in-memory, so restart means state is lost
  - This means we need to kill all existing lessons on start
- Fairly opaque

NRE LABS

MP - Syringe Redesign

- Better resilience - no “one syringe”
- Easier to extend - just pop a new service on the message bus
- Easier to reason about, maintain, and contribute to individual services



NRE LABS

User Experience

- ✔ Are users having problems?
  - Monitoring components is easy, monitoring the full thing is hard?
- ✔ If they are, what can we even do about it?
  - By definition, our users aren't experts in Github
  - In the 0.01% of cases where users find a way to get feedback to us, all of the context is lost.

NRE LABS

MP - Observability Instrumentation

- ✔ **User feedback right in the UI**. Click this button or type in this box to tell us about a problem. Responses go to some kind of queue for filtering and triage. Includes session and request IDs
- ✔ **Better centralized and structured logging**
- ✔ **System observability** - Tracing from web front-end all the way through every syringe microservice. HIgh cardinality based from initial session and lesson ID allows us to get to a specific interaction easily.

MP - Standard and Secure Endpoint Images

- ✔ Target: security of VMs with developer experience of containers
- ✔ Candidates:
  - Weave Ignite
  - Kata Containers
  - Custom tooling
- ✔ KubeVirt
  - May help in future but this is mostly focused on ops-side stuff we don't need.

NRE LABS

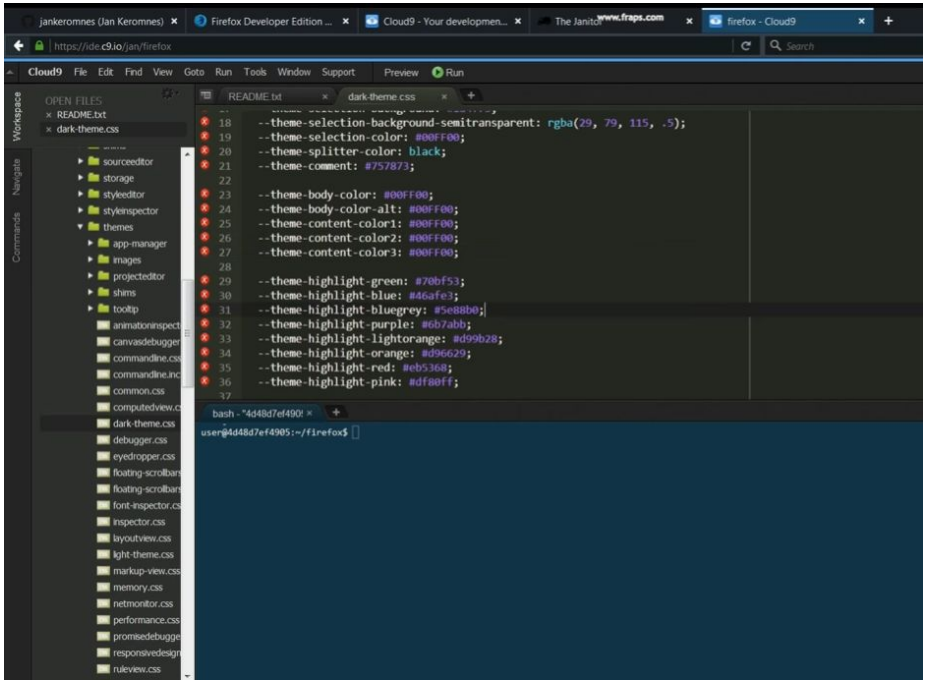
MP Code Name “Matt Is Not A Web Dev”

- ✔ Hiring open source web dev firm to do a UX review and give us a new base
- ✔ Like……MOBILE SUPPORT
- ✔ Will still have a ton of work left for the community to do

NRE LABS

MP - Build Lessons in the Browser

- ✔ Come full-circle and enable lesson contributions in the browser
- ✔ Using Mozilla Janitor as inspiration (uses Amazon Cloud9)
- ✔ Have to figure out a cost-effective deployment model



NRE LABS

Resources

Labs - labs.networkreliability.engineering

Community - community.networkreliability.engineering

Open Source - github.com/nre-learning

Antidote Docs - antidoteproject.rtfd.io

Standups - Every Tuesday 8AM Pacific

Twitter - @NRELabs

No Contribution Too Small!

- Use NRE Labs and open issues!
- Lesson Contributions - new or existing
- Platform enhancements/fixes

NRE LABS

Questions?

NRE LABS