KubeCon

CloudNativeCon

North America 2019

Moving from Legacy Infrastructure to the Cloud in a Government Organization

Chris Carty, CKA, Senior Developer / Integrator, City of Ottawa

Capital of Canada

I live here

Temperature

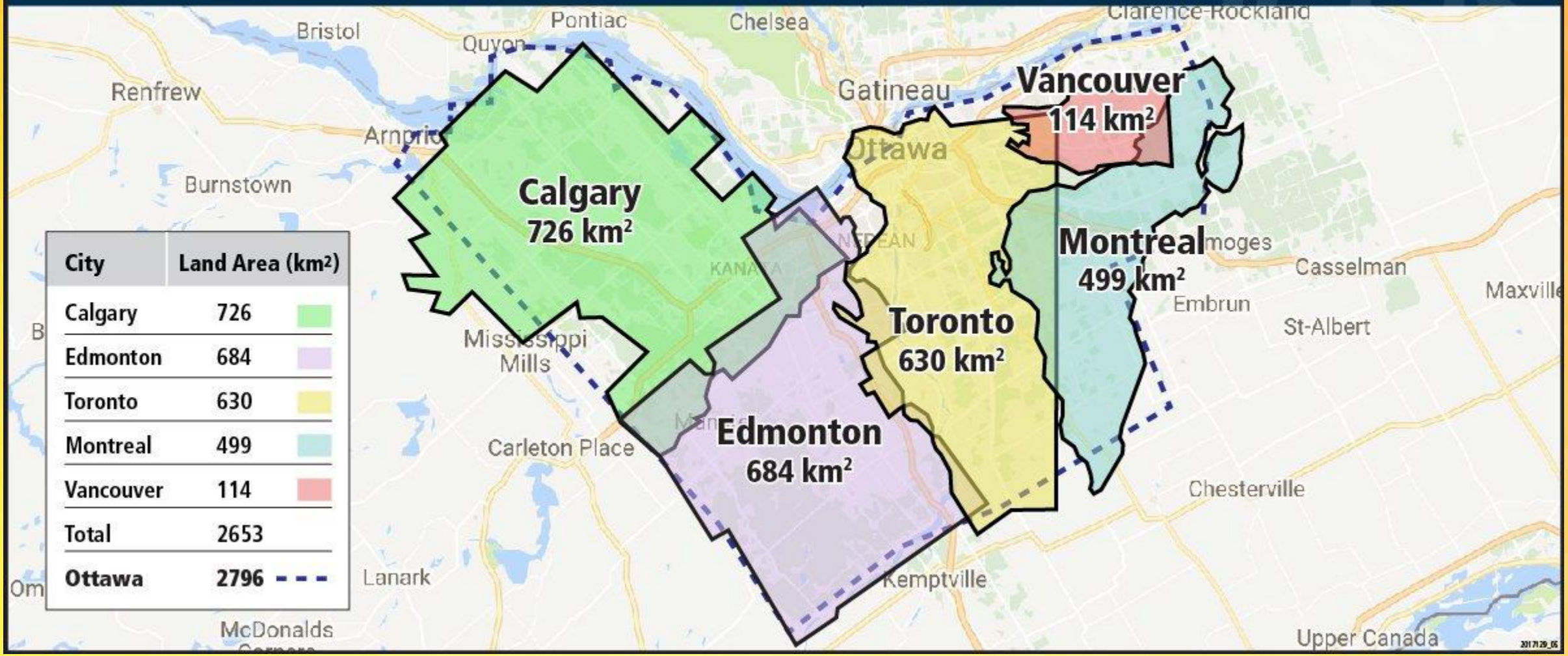
- Record low with windchill -47.8C (-54F)
- Record high with humidex +47.2 (+117F)

-47C --> +47C

Size

- Population just hit 1 million
- In 2001, 11 cities and townships amalgamated
- Ottawa is big: 2,796km2 (1737mi2)

Ottawa's scale
in perspective



City	Land Area (km²)
Calgary	726
Edmonton	684
Toronto	630
Montreal	499
Vancouver	114
Total	2653
Ottawa	**2796**

City of Ottawa Complexity

Supports for federal government workers, addresses urban and rural concerns, and maintains infrastructure through temperature extremes

- 17,000+ employees (300+ IT)
- 120+ different lines of business
- 400+ Applications (CoTs, Java, dotNet, Perl)

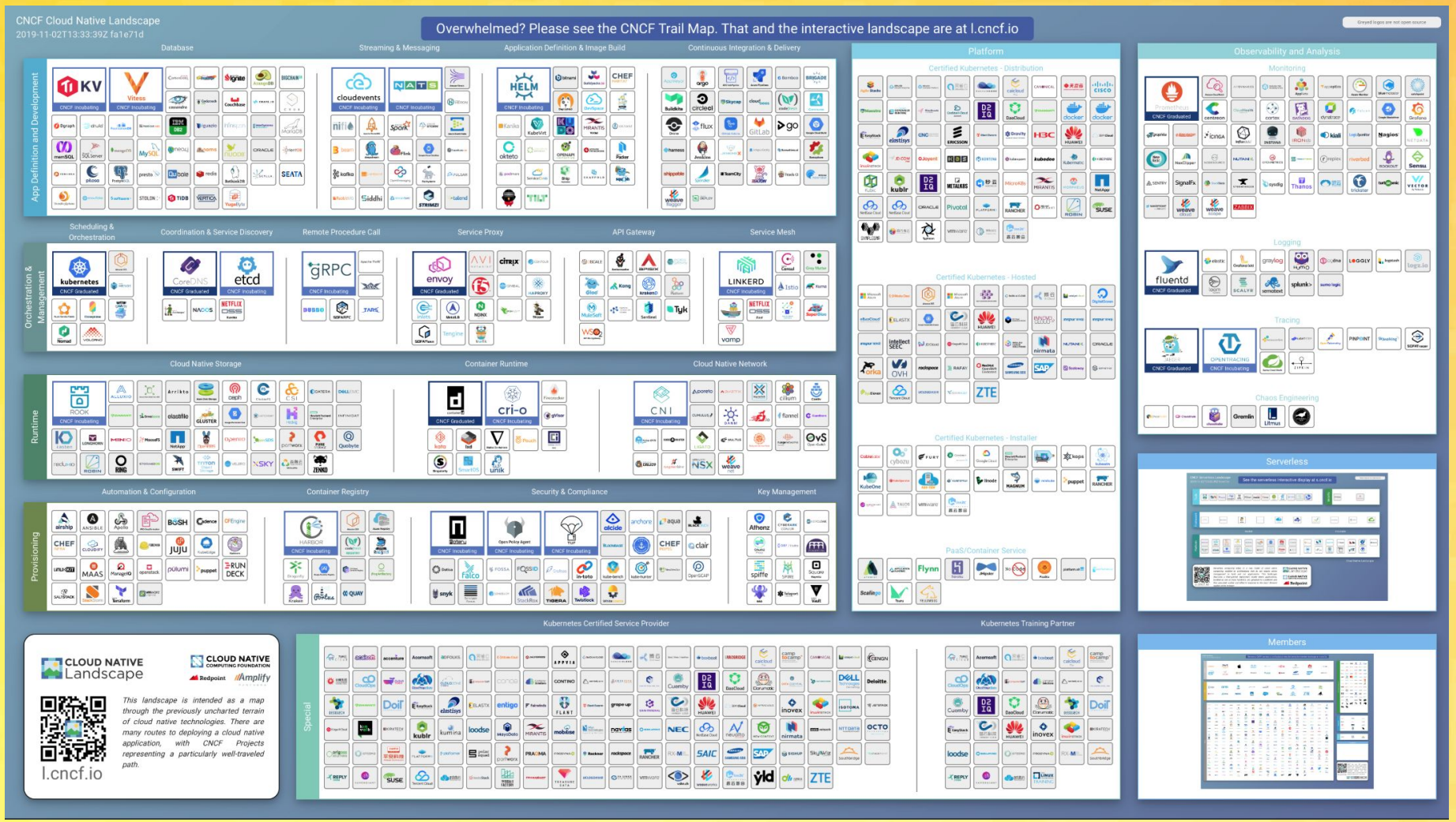
Part 1:
How to Do Kubernetes Stuff

Who, Where, How to Deploy

CNCF Landscape

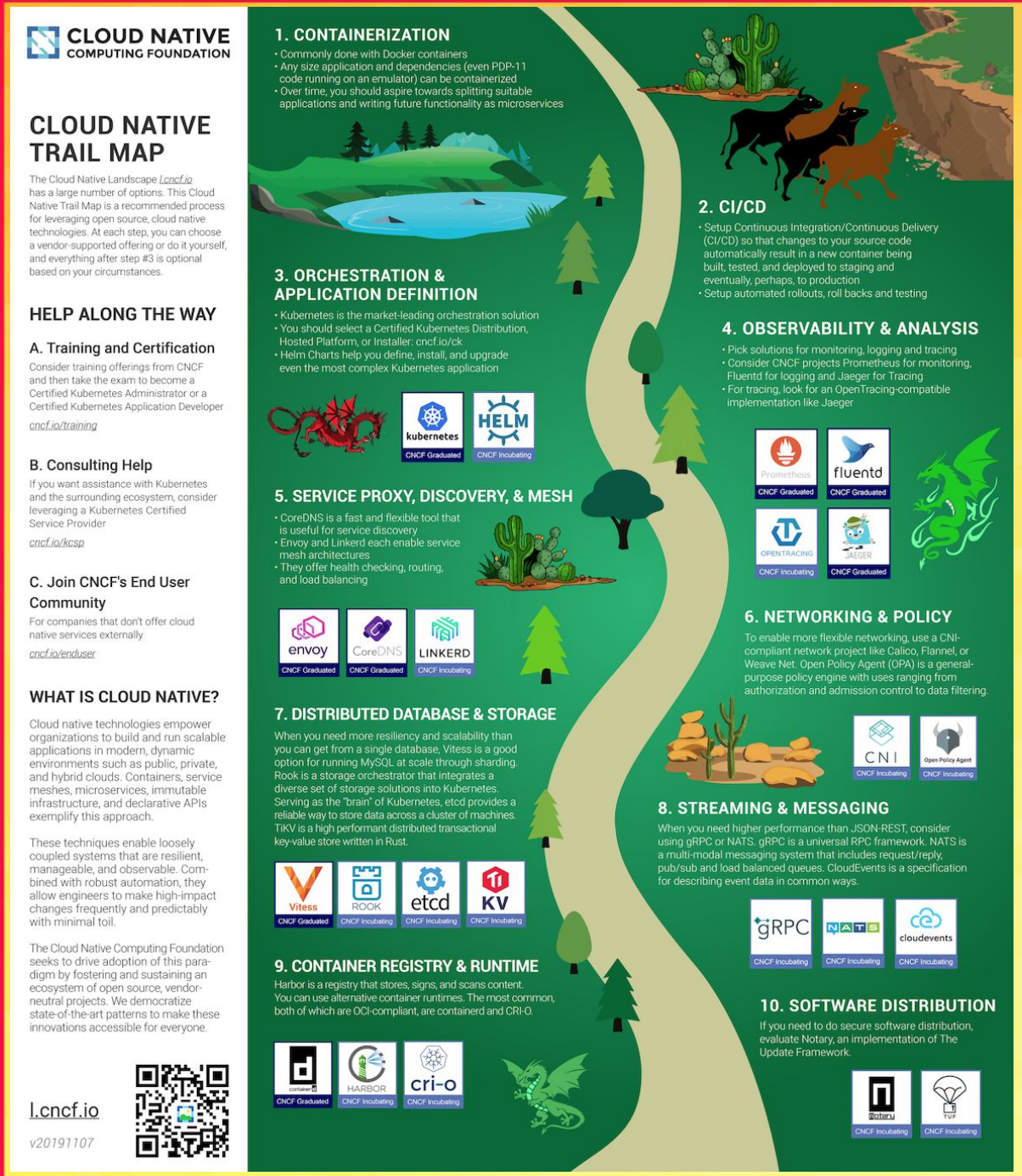
KubeCon | CloudNativeCon

North America 2019



Break Things Down

1. K8s Distribution (Cluster Management)
2. Security
3. Storage
4. CI/CD
5. Container Registry
6. Monitoring
7. Key Management
8. Ingress

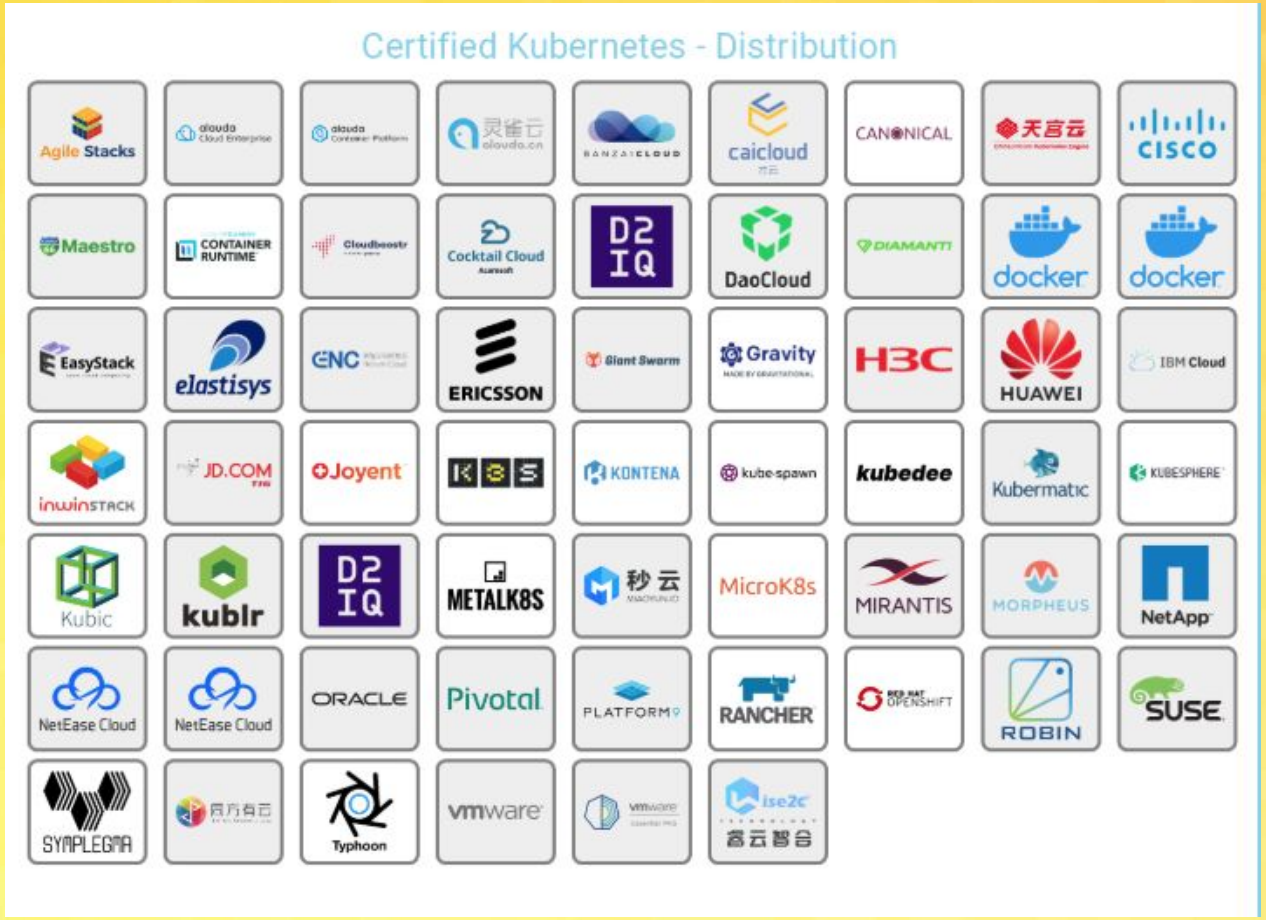


KubeCon | CloudNativeCon

North America 2019

1. Containerization
2. CI/CD
3. Orchestration & Application Definition
4. Observability & Analysis
5. Service Proxy, Discovery, & Mesh
6. Networking & Policy
7. Distributed Database & Storage
8. Streaming & Messaging
9. Container Registry & Runtime
10. Software Distribution

Selecting a Distribution



Setting Objectives

- **Vendor Agnostic** - keep options open for evolving technology
- **Open Source Kubernetes**
- **On-Prem and Public Cloud** - sensitive workloads on-prem
- **Multiple Clusters** - security blast radius
- **Enterprise Level Support** - no team with knowledge onsite
- **Future Support for Windows Containers**
- **Role Based Access Controls** - granular security controls

Try Things Out

- Use open source software to test before commitment
- Leverage the tools you already have
- Try trial demos (Vendor Hall)

Allows for a detailed Request For Proposal or Sole Source

Patterns Over Tools

Tools should be swappable to give the flexibility to meet the changing needs of various teams and new tools available

People → Process → Technology

Our Stack

- **Orchestration**: Kubernetes, Rancher, Azure Kubernetes Service
- **Observability**: Prometheus/Grafana, ELK Stack
- **Security**: Open Policy Agent, Network Policies, Kube-Bench/Hunter
- **Service Mesh**: Linkerd
- **CI/CD**: Fluxcd, GitLab, Azure DevOps
- **Registry**: Harbor
- **Storage**: NetApp Trident, Longhorn
- **Ingress**: ingress-nginx

Part 2:
How to Get People to Use It

The Hard Part

Growing Awareness

- Lunch presentations
- Senior management presentations
- Team presentations
- One-on-one sessions
- Phippy and Friends

Full adoptions requires massive cultural shifts in legacy organizations.
Go slow, be patient, and get it right.

Finding Pain Points

Meet with project teams to discover specific pain points

Offer carrots not sticks

- Scalability - ability to spin up new instances without pre-provisioning servers
- Self healing - reduced on call

Automate All of the Things

Automate where possible to minimize knowledge needed to deploy to Kubernetes. Not all users need to be experts.

- Create starter templates that follow best practices
- Continuously iterate for easier adoption as issues are identified

Migrating Applications

- Legacy Perl and Java apps
- Migrating from Unix To Kubernetes
- Similar Build/Deploy Process

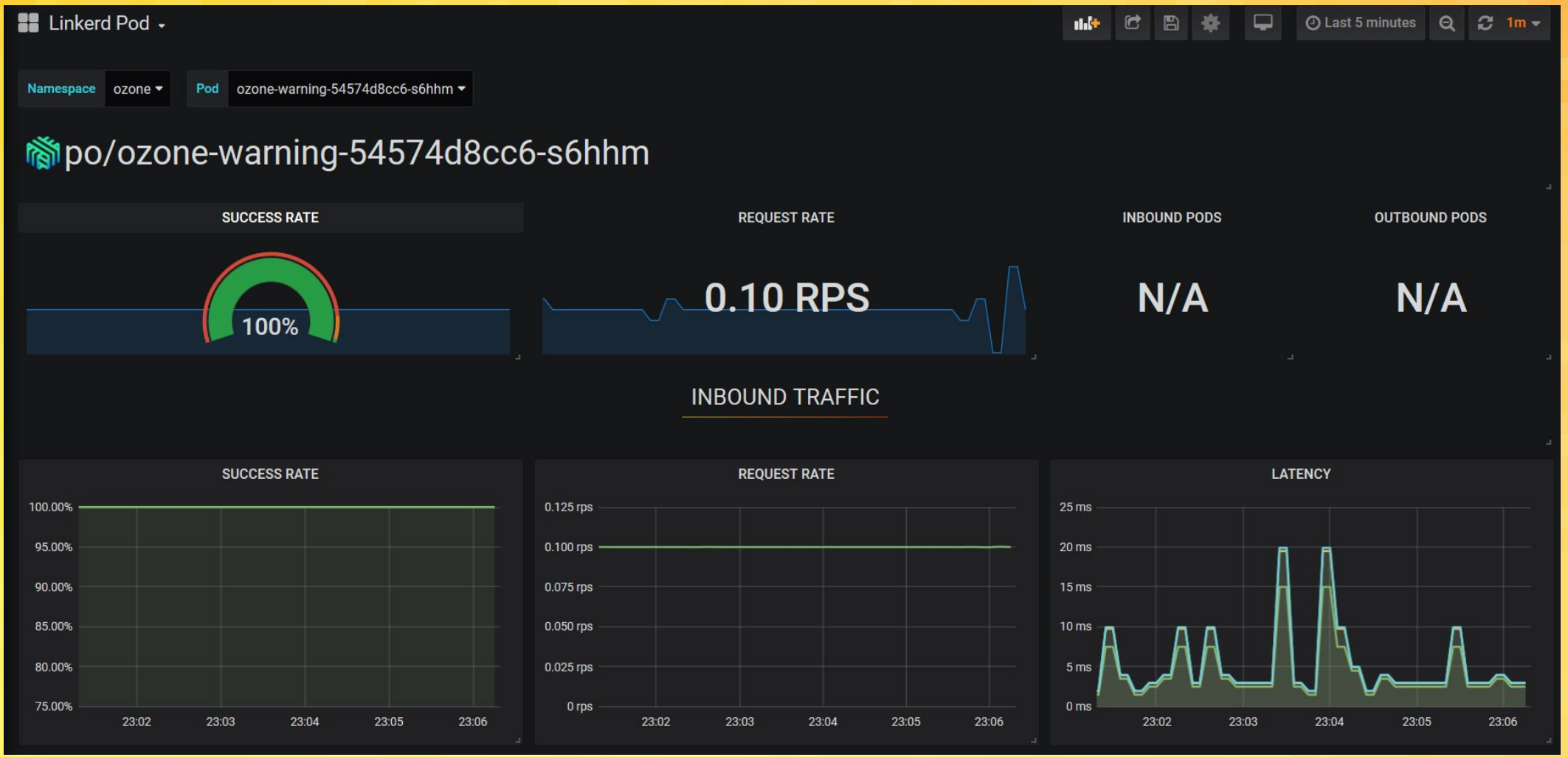
Drag and Drop Deploy

CI/CD configuration | Auto DevOps enabled | Add Kubernetes cluster

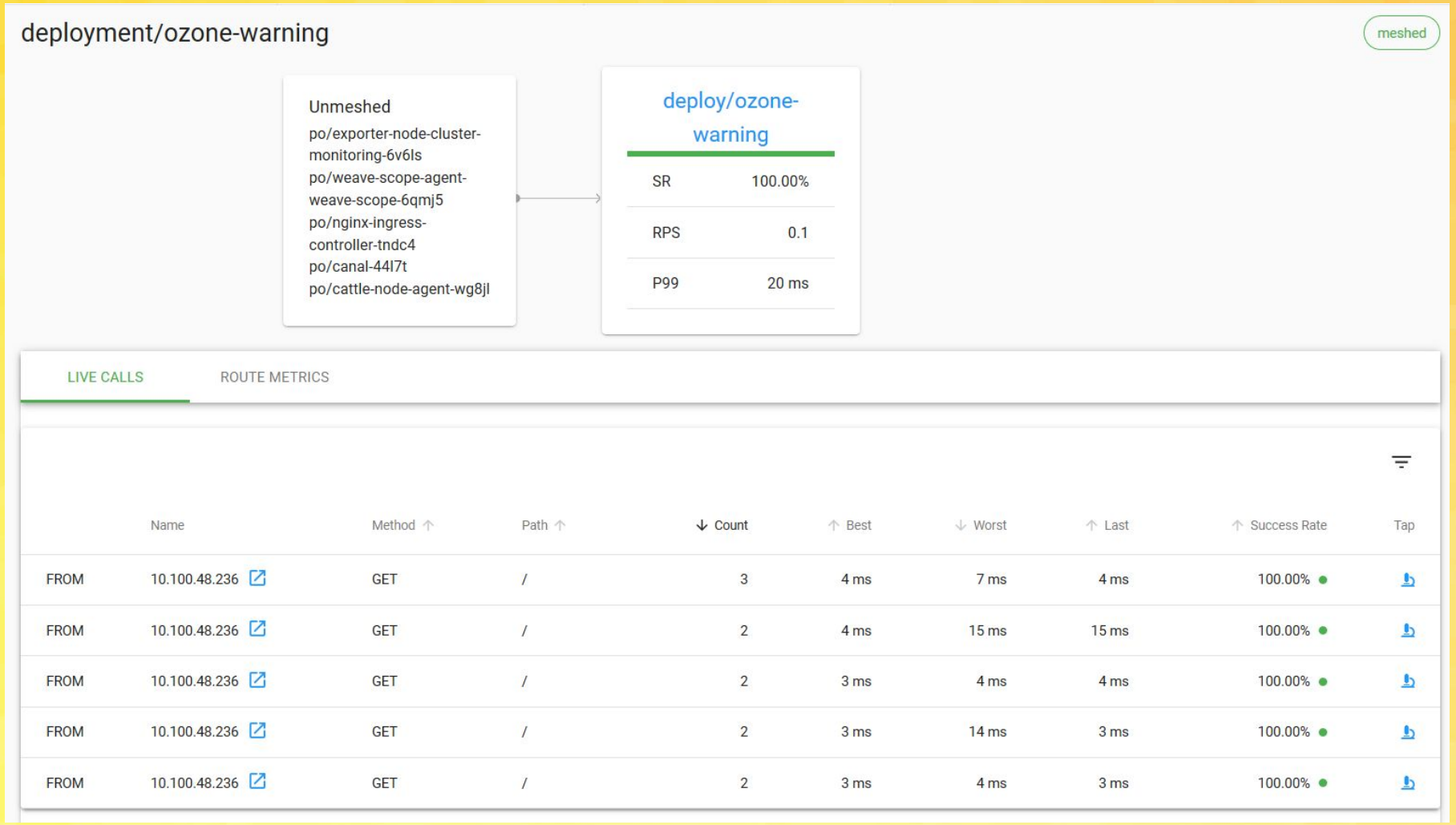
Name	Last commit	Last update
.gitlab-ci.yml	updated project name	2 days ago
Dockerfile	first commit	2 months ago
graffiti.war	Replace graffiti.war	1 month ago

Added Metrics

Traffic


Linkerd Pod
Last 5 minutes
1m
Namespace
ozone
Pod
ozone-warning-54574d8cc6-s6hhm
po/ozone-warning-54574d8cc6-s6hhm
SUCCESS RATE
100%
REQUEST RATE
0.10 RPS
INBOUND PODS
N/A
OUTBOUND PODS
N/A
INBOUND TRAFFIC
SUCCESS RATE
100.00%
95.00%
90.00%
85.00%
80.00%
75.00%
23:02
23:03
23:04
23:05
23:06
REQUEST RATE
0.125 rps
0.100 rps
0.075 rps
0.050 rps
0.025 rps
0 rps
LATENCY
25 ms
20 ms
15 ms
10 ms
5 ms
0 ms

Linkerd


deployment/ozone-warning
meshed
Unmeshed
po/exporter-node-cluster-monitoring-6v6ls
po/weave-scope-agent-weave-scope-6qmj5
po/nginx-ingress-controller-tndc4
po/canal-44l7t
po/cattle-node-agent-wg8jl
deploy/ozone-warning
SR 100.00%
RPS 0.1
P99 20 ms
LIVE CALLS
ROUTE METRICS
Name Method Path Count Best Worst Last Success Rate Tap
FROM 10.100.48.236 GET / 3 4 ms 7 ms 4 ms 100.00%
FROM 10.100.48.236 GET / 2 4 ms 15 ms 15 ms 100.00%
FROM 10.100.48.236 GET / 2 3 ms 4 ms 4 ms 100.00%
FROM 10.100.48.236 GET / 2 3 ms 14 ms 3 ms 100.00%
FROM 10.100.48.236 GET / 2 3 ms 4 ms 3 ms 100.00%

Training

Provide tools and guidance for developers to learn at their own pace

- One-on-one and team mentoring
- Pluralsight
- Katacoda
- CNCF Kubernetes Fundamentals
- Kubernetes.io/tasks
- Kubernetes the Hard Way

Conclusion

Current State - If You Build It...

- Internal Dev, QA and Prod and External QA and Prod clusters
- 3 Teams are engaged and actively developing + others testing
- First application with public traffic in prod
- Azure App Service and VMs are competing systems in use

Looking Ahead

- First Java Applications will be going live for external traffic
- Internal web forms will be served internally
- Corporate Container Security Standards will be approved
- Service accounts setup and cloud governance
- Increased automation tooling (ie Terraform)

Key Takeaways

1. Set objectives
2. Work through your pathway
3. Develop patterns
4. Patiently spread awareness
5. Dangle “carrots”
6. Automate all the things
7. Come back next year

Resources / Contact

- github.com/cartych/kube-con-resources
- Follow me @macintoshprime
- email: christopher.carty@ottawa.ca

Questions / Discussion