

# Network Automation Evolution

From cowboy-style scripts to highly orchestrated networks

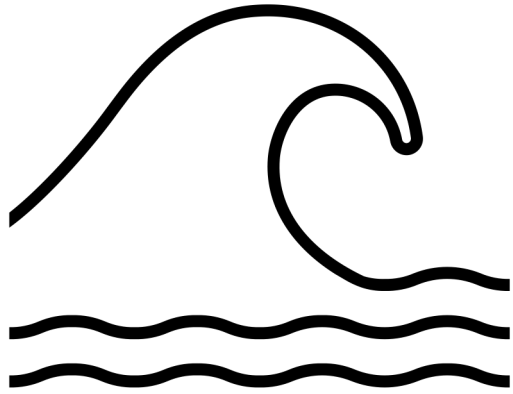
Diogo Montagner

Sr Network Architect

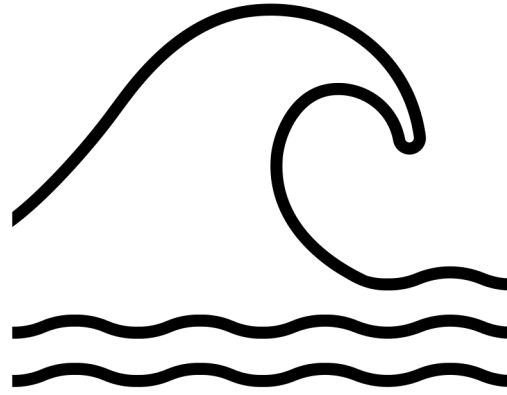
# Agenda

- The Automation Journey
- The Integration Pain
- We forgot the Operational people
- Messages from far, far away
- Where to go from here

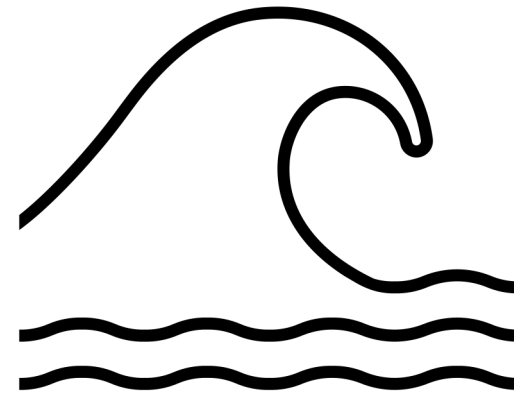
# The Automation Journey



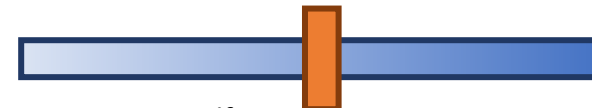
AI/ML



Telemetry

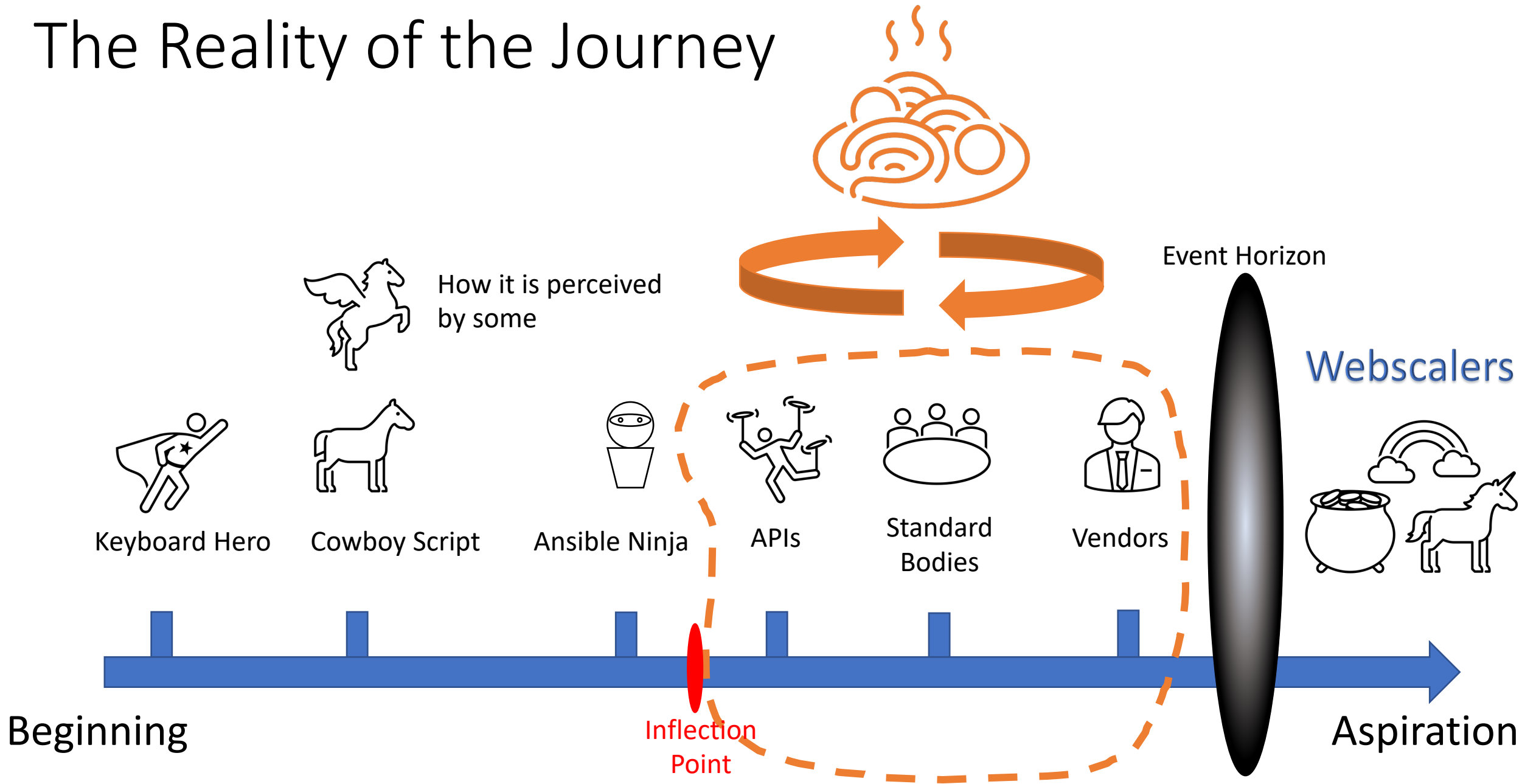


Automation

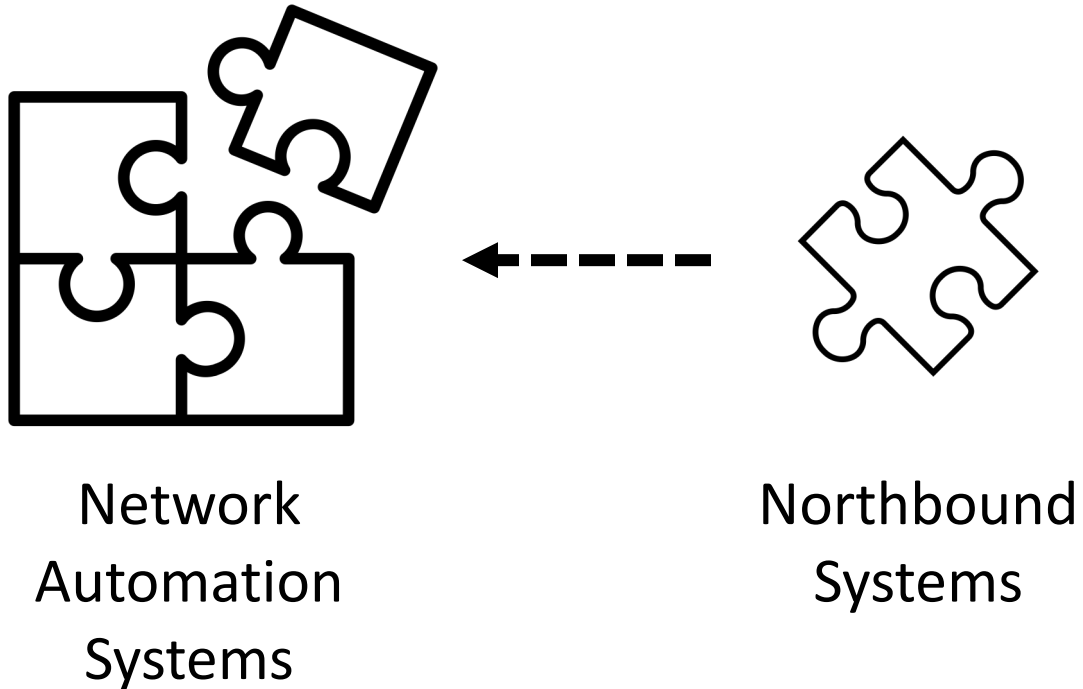


Low High

# The Reality of the Journey



# The Integration Pain



## Few Tough Design Questions/Decisions

- Q** - Common API interfaces (standards driven)
- Q** - API Translation (API gateway)
- Q** - Data models: my own model vs standards driven
- Q** - Buy vs Build

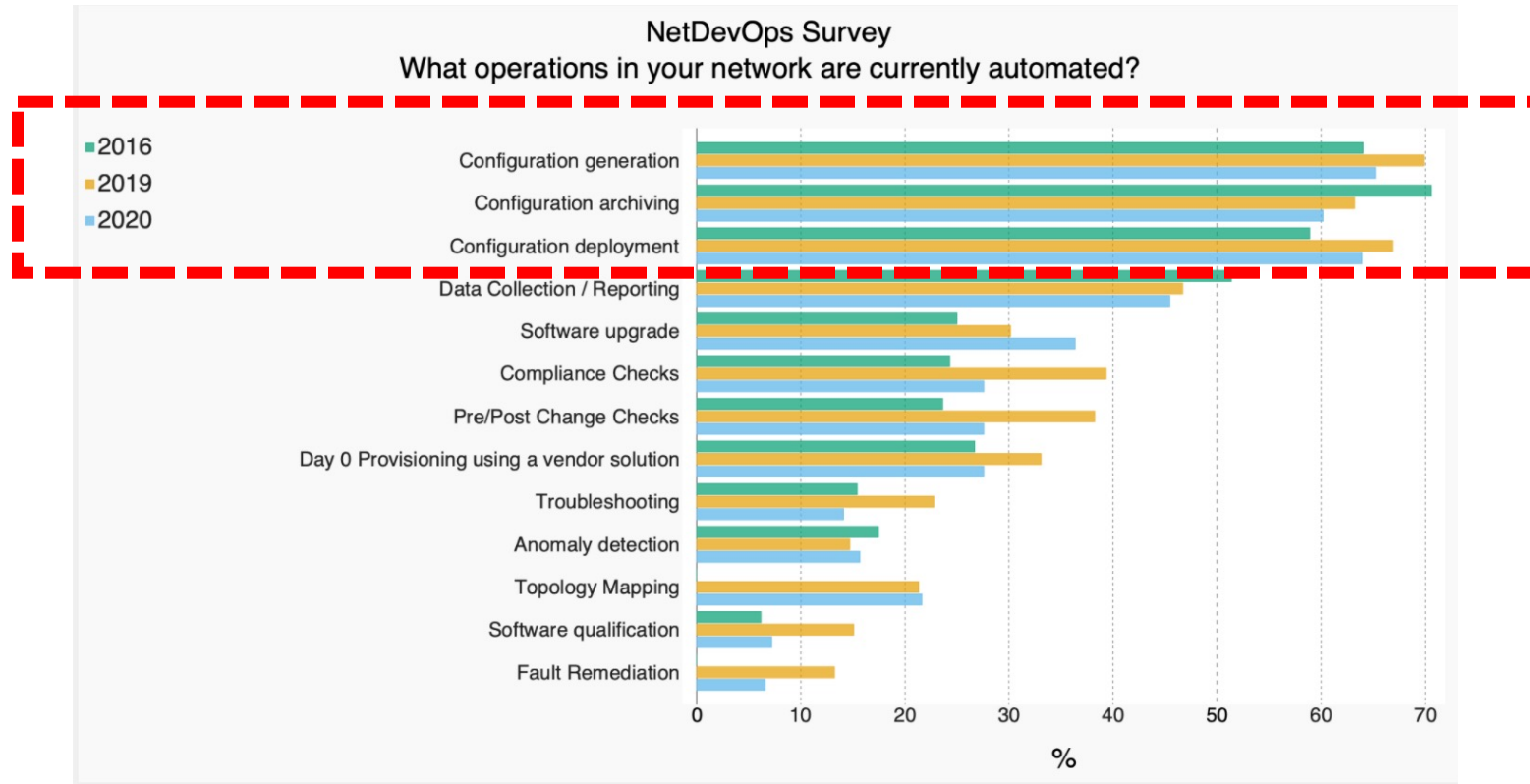
**Answer: it depends !!!**



# We forgot the operational people

- Fulfillment is the low hanging fruit of automation
- Day-to-day operations is a secondary thought (closed-loop)

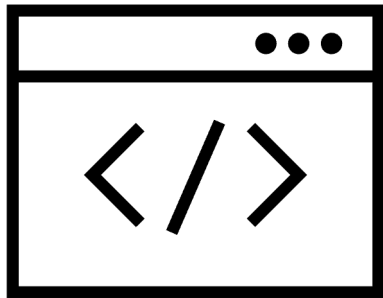
Automation  
Complexity



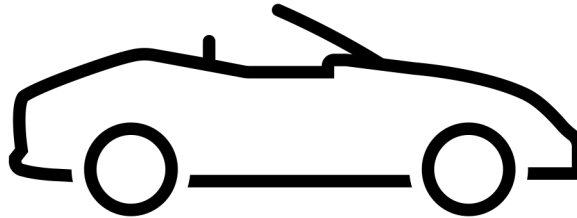
# The Hard Question For Us

( Us == the network industry )

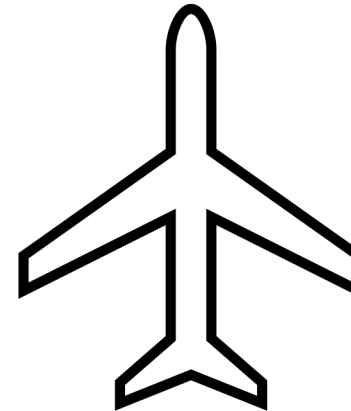
**What does it take for us to level up the automation game ?**



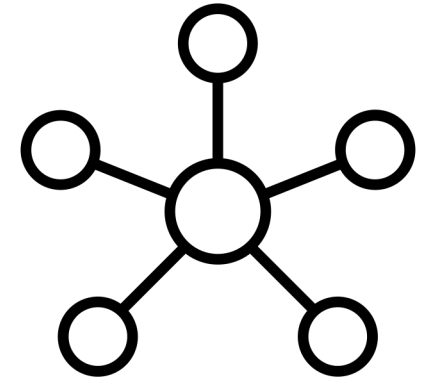
Web industry



Car industry

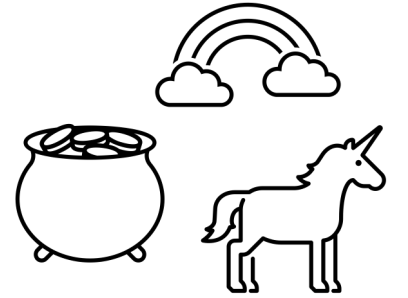


Aviation industry



Network industry

# Welcome to the overhyped “SD” world



- Rush towards the pot of gold at the end of the rainbow (OpenFlow, P4)
- Software-Defined { Network | DC | Access | < whatever sounds cool > }
- It was the solution for every problem under the sun
- ML/AI (for network) is having similar rush these days



# SDN has come a long way

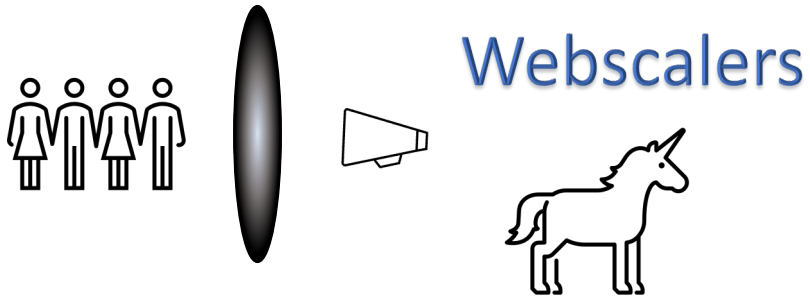


- Moved away from solving everything in one place
- Tailored controller-driven solutions for specific problem spaces



- One controller for each problem
- Coordination among multiple controllers

# Messages from far, far away



- Walk away from the keyboard
- Manage cattle instead of pets
- Architect for automation

# Decoding the message ...

Walk away from the keyboard

- Layered abstraction approach
- Less about knobs and config – automate for outcomes
  - Instead of “**configure one BGP session to each route reflector with these knobs**”
  - Focus on “**configure route reflector peering**”

# Decoding the message ...

Manage cattle instead of pets

- Design the automation for **scale**
- Design the automation to be **fungible**
- **Be careful** automating human-like workflows
- Keep the **blast** radius small
- Things will **fail**: an individual failure will not compromise the group

# Decoding the message ...

## Architect for automation

- Network architecture and good automation go hand-in-hand
- Apply K.I.S.S principle
- Loose coupling
- Horizontal scale
- Understand your failure domains and their respective SLAs
- Implement safety nets
  - A.K.A.: fail-safe, big red stop button, break-glass, checks and balances

# How we get to the other side ?



- Not a technological problem (many think it is)
- Balance between strategic vs tactical
- Business, Technical and Management need to come together
- Environment and culture
- Embrace change, embrace failure
- Celebrate success

# Where to go from here ?

- Know where you are and what your goals and timelines are
- Create a strategy, develop a plan, execute, monitor, adapt
- As an industry, we need to:
  - think automation first
  - make automation seamless
  - simplify integration
  - remove the noise
  - make automation sustainable

Thank You!