



# The Network Automation Forum's Network Automation Framework

Jeff Doyle  
Data Center and Automation Architect  
CHI-NOG 13

The Gold Standard  
in Cybersecurity and Network Engineering



# Agenda

- Drivers for Network Automation
- Barriers to Network Automation
- Approaches to Network Automation
- The Network Automation Framework

# Drivers for Network Automation

- Network Complexity
  - Too complex to run from a CLI
  - An infrastructure needs to be seen as a holistic entity
  - Consistency is essential
  - Too much information to process!
- Time is Money
  - OPEX is usually the largest part of a CIO's budget
  - Operators should be looking at “the big picture”
  - Architects should not be firefighters
- **Humans!**
  - 60%-80% of network outages are caused by human error\*
    - Simple typing mistakes
    - Failure to follow procedures
    - Poor or non-existent policies and procedures

\*The Uptime Institute, Annual Outage Analysis Report 2025

# Barriers to Automation

- Automation Complexity
  - Where do I start?
  - Where do I get the talent?
  - How do I test it?
  - NOS diversity
  - Poor definitions
    - (If you can't quantify something you haven't asked enough questions)
- Cultural Resistance
  - Automation will eat my job!
  - What's the business case?
- Fear!
  - Losing control of my network
  - Automating a broken process

# Approaches to Automation

## DIY

- OPEX intensive
- Slow development time
- Ongoing maintenance
- Custom built to your network, your objectives

## Open Source

- Free, sort of...
- Large community support
- Proven, usually...
- Customizable

## COTS

- CAPEX intensive
- Fast deployment
- Easy maintenance
- Bells and whistles you might not want

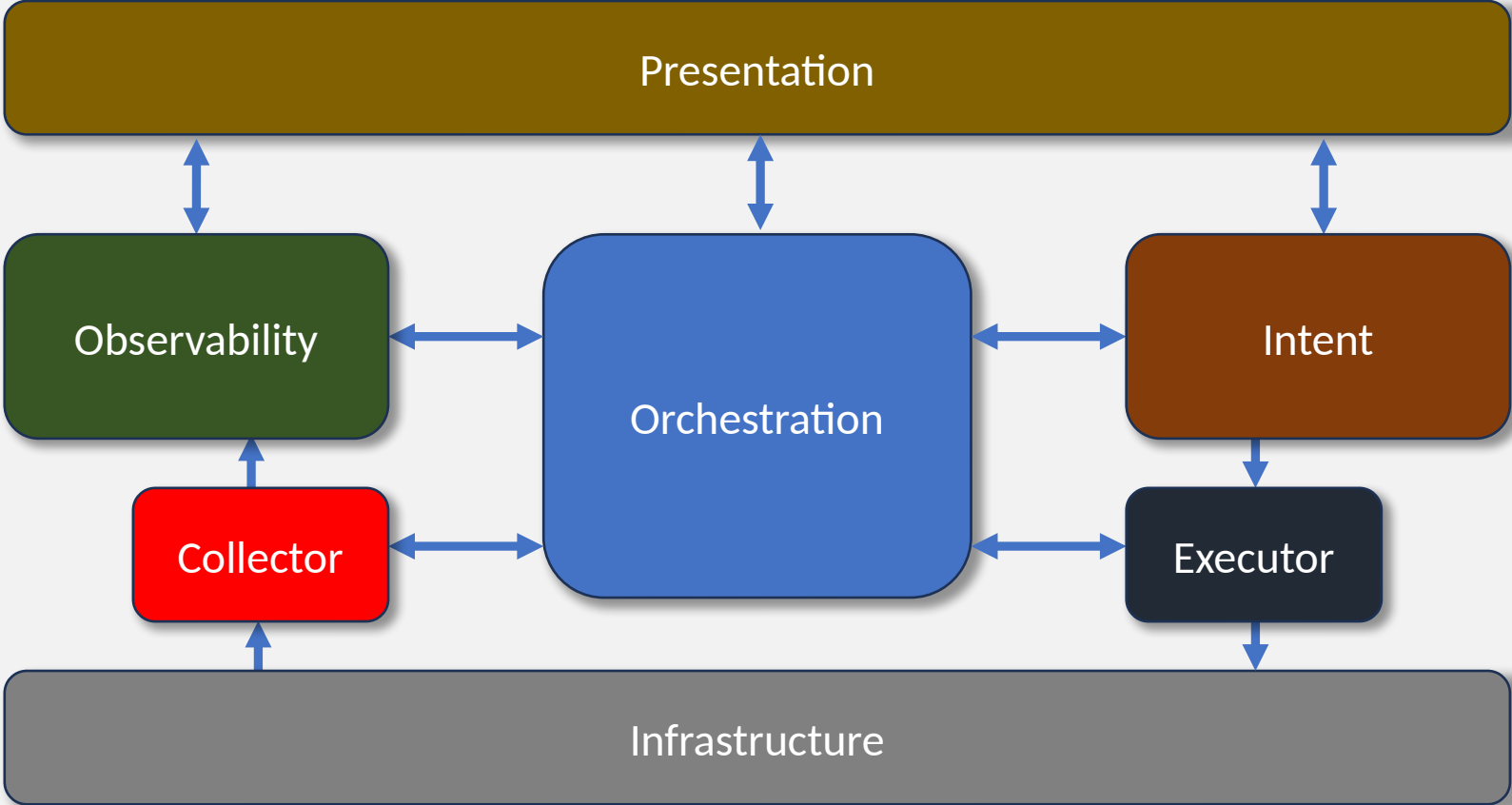
## Hybrid

- Combining expertise
- Moderate expense
- Develop what you can, buy what you must
- Allows gradual development

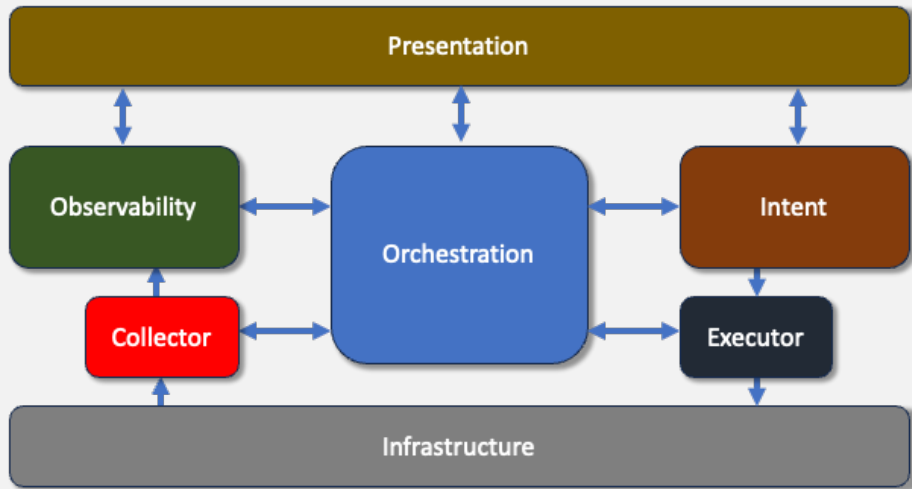
# What is the Network Automation Framework?

- A modular, vendor-neutral reference design for automation
  - Design
  - Implementation
  - Refinement
- Defines key functional building blocks
- Relevant to both new deployments and for evolving solutions
- Adaptable to CI/CD methodologies

# Framework Architecture



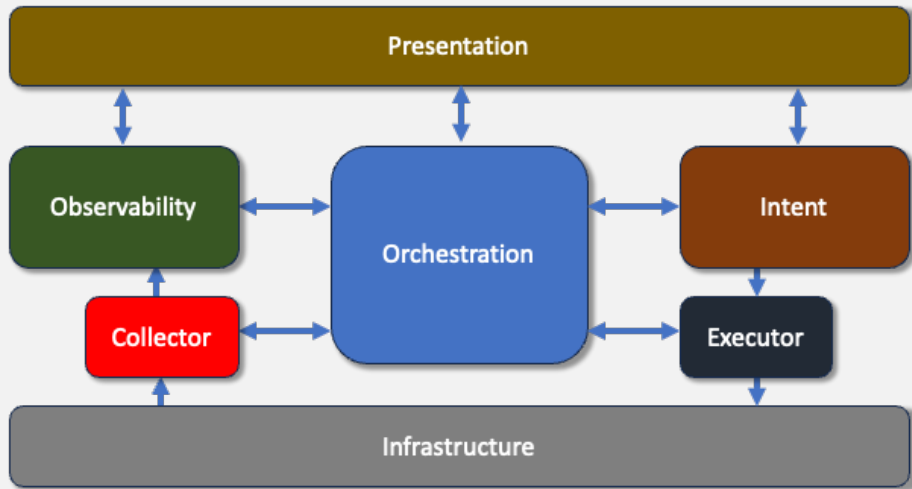
# Framework Architecture



## Collector

- Reads the state of the infrastructure
- Read methodology

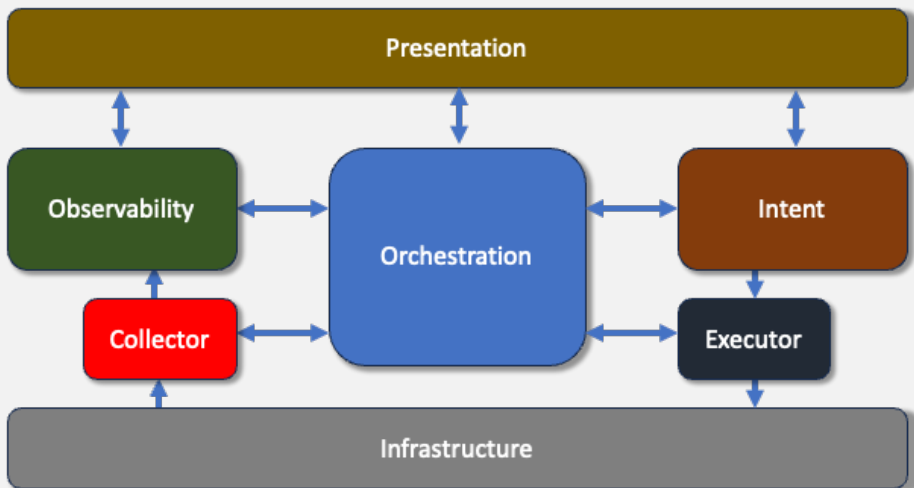
# Framework Architecture



## Observability

- What I HAVE
- A database of observed state
- Observational logic and analytics
- Normalized
- A SSoT?

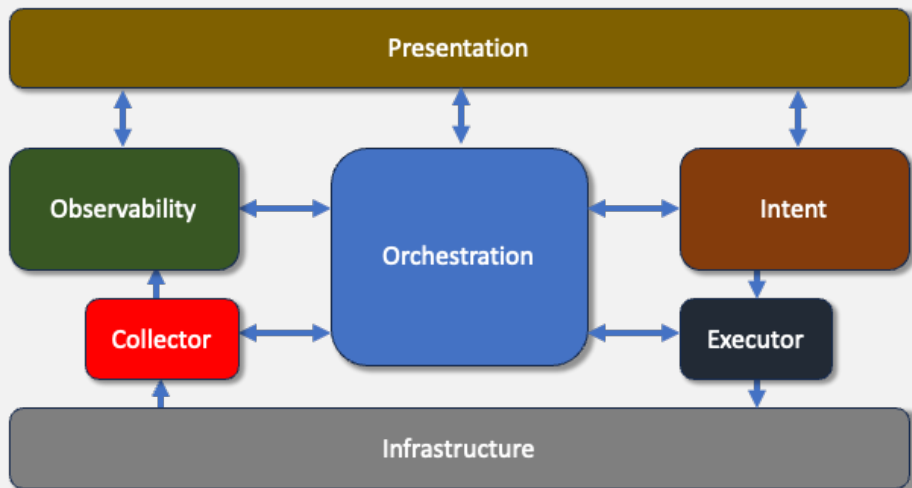
# Framework Architecture



## Intent

- What I WANT
- The desired state of the network
- Both:
  - Configuration
  - Operational expectations
- Applied to an *abstraction* of the physical infrastructure

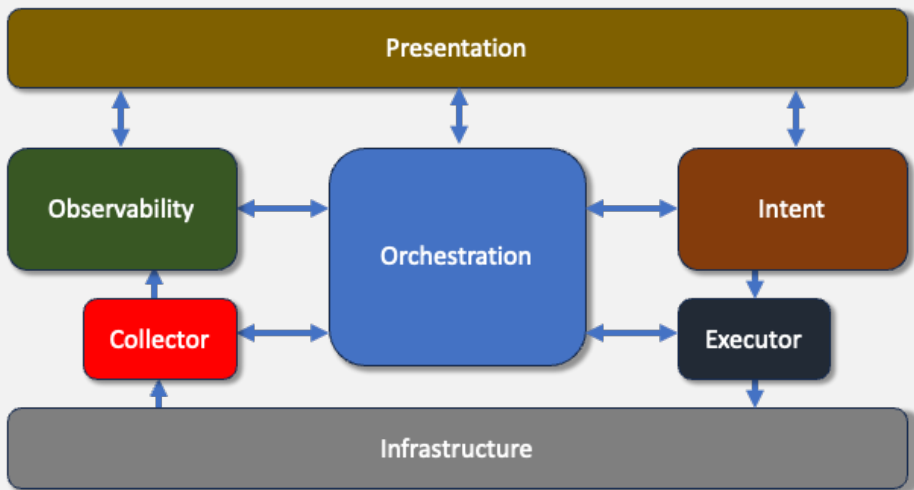
# Framework Architecture



## Executor

- Writes to the infrastructure
- Translation from abstracted intent database to vendor-specific commands?

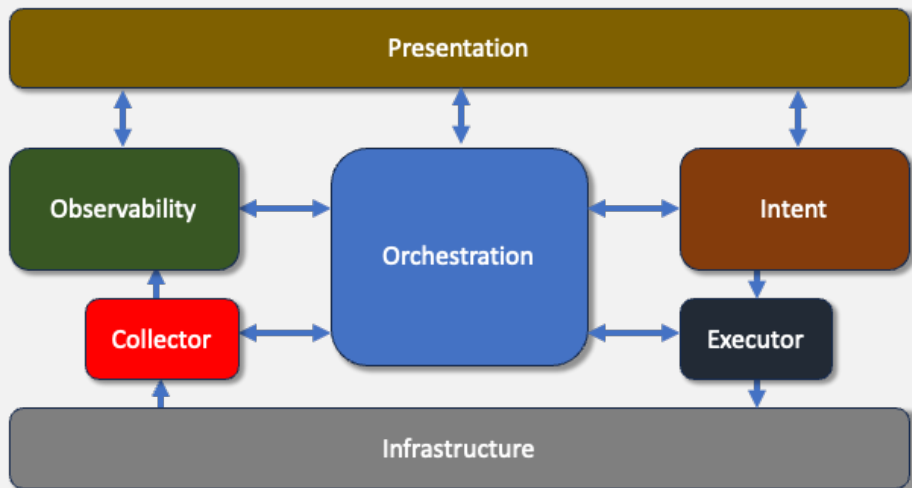
# Framework Architecture



## Orchestration

- Ties simple automation tasks into a workflow
- Scheduling
- Events
- Rollback
- A dry-run function?
- May or may not interact directly with Collector and Executor

# Framework Architecture



## Presentation

- User interface
- Not necessarily "a single pane of glass"
- Includes:
  - Dashboards
  - GUIs
  - CLI tools
  - API gateways
  - AI interfaces

# A Work Still in Progress

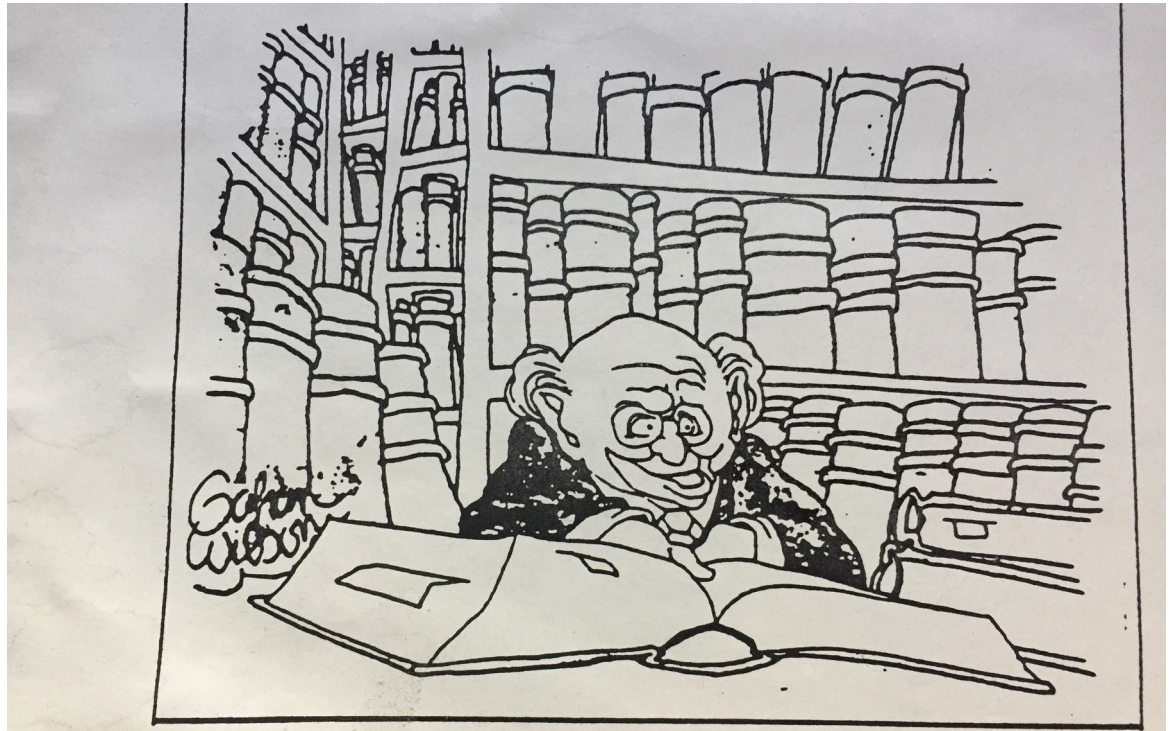
- The NAF is still evolving the framework
- User feedback is welcomed and encouraged
- Documentation is continuously updated on the NAF website and in Github

<https://networkautomation.forum/>

<https://github.com/Network-Automation-Forum/reference/tree/main/docs/Framework>

[jdoyle@agn.tech](mailto:jdoyle@agn.tech)

# Questions?



"By golly, for a minute there it suddenly all made sense!"