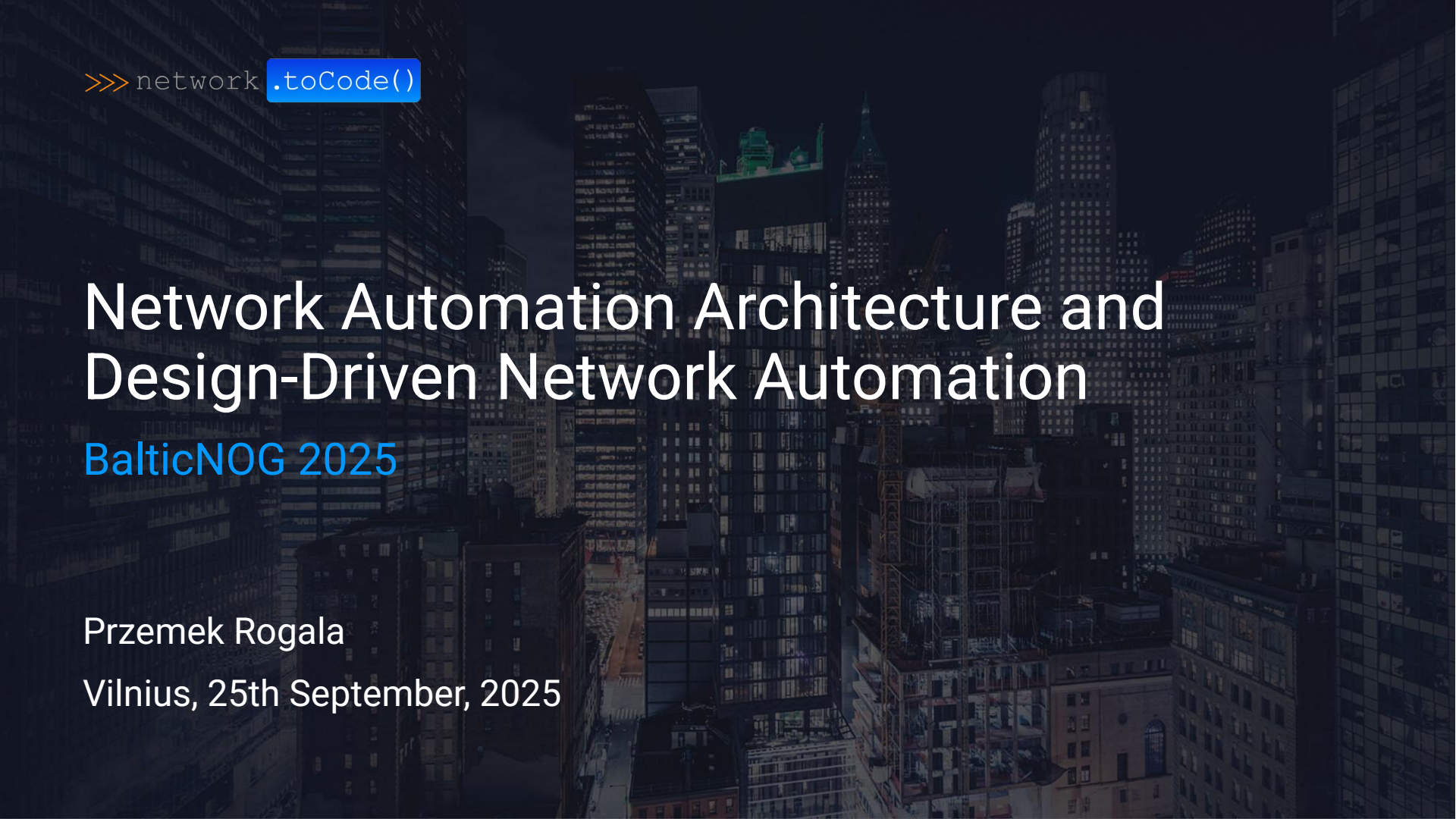


Network Automation Architecture and Design-Driven Network Automation

BalticNOG 2025

Przemek Rogala

Vilnius, 25th September, 2025

>>> About me

Przemek Rogala

Network Automation Architect

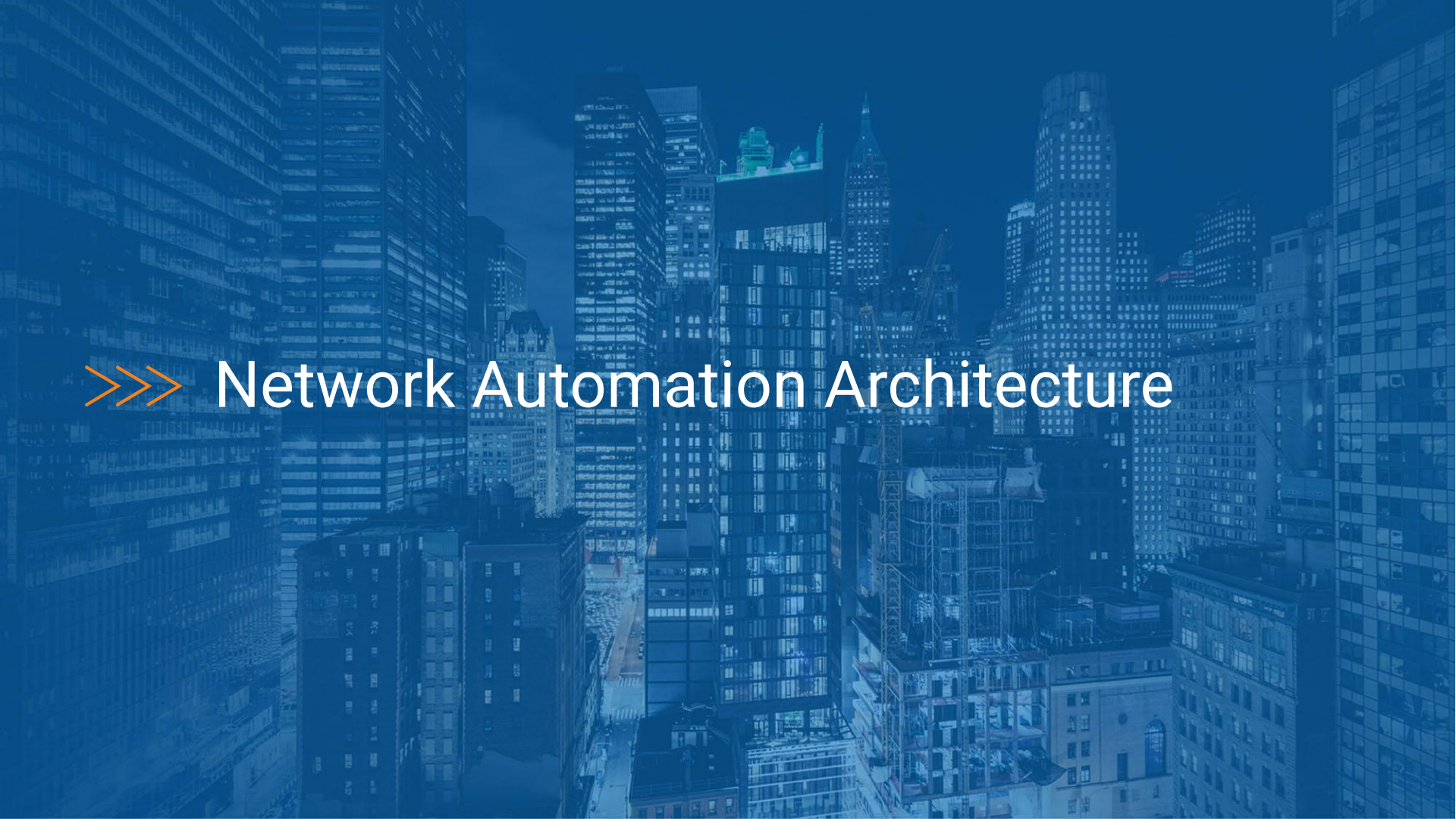
in/przemek-rogala

>>> network .toCode()

Network to Code designs, builds, deploys, and manages network automation platforms that provide open, extensible, and vendor agnostic network automation solutions powered by a source of truth.

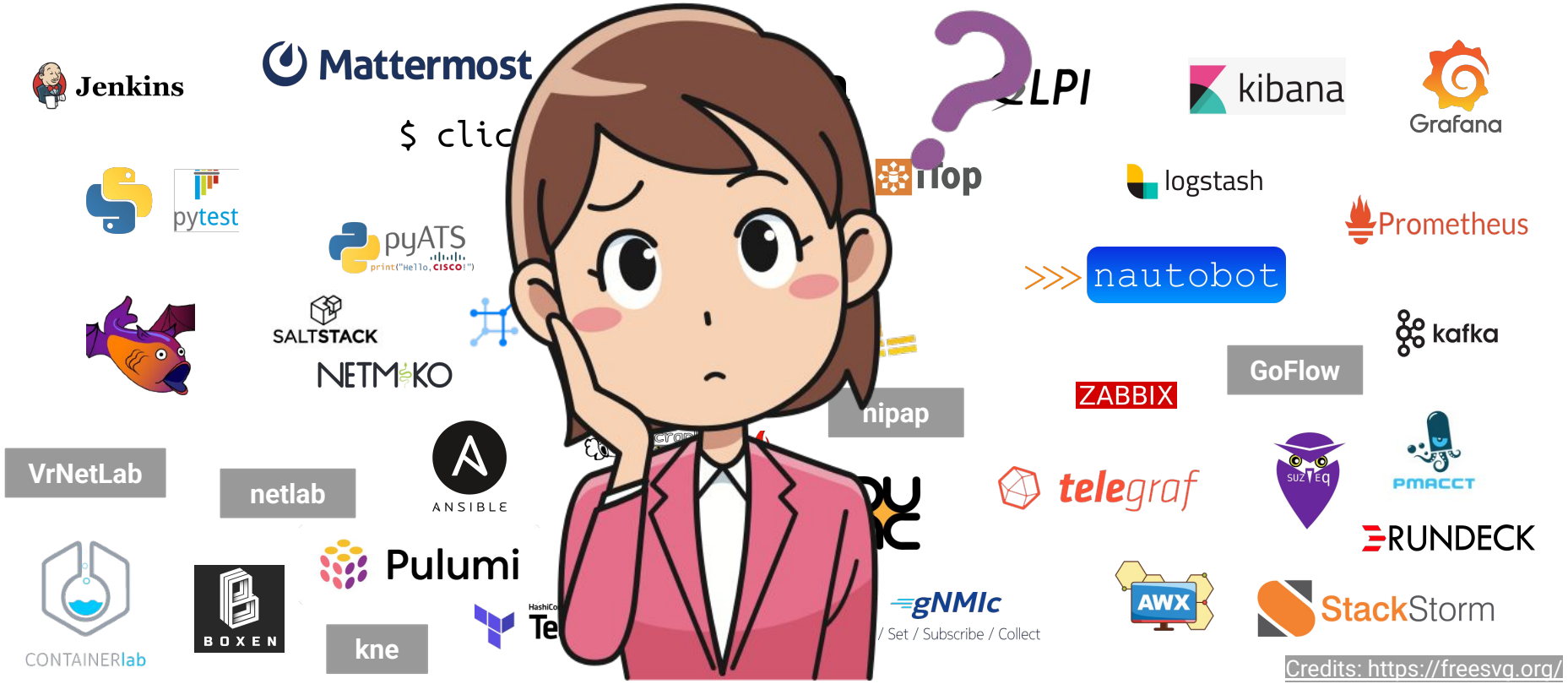
Based on an original talk by Christian Adell

in/christianadell



Network Automation Architecture

>>> Network automation solution tooling options



Credits: https://freesvg.org/

>>> What is the goal?

OS Upgrades

Series of automation that allow the *safe* upgrading of network devices, firewalls, and wireless controllers.

Firewall Rule Automation

Deployment of firewall rules for from an ITSM tool or other automation system.

Switchport Configuration

Drive switchport configuration changes by updating the database and pushing configurations.

SoT Data Sync

Synchronize, unify, and enrich network data sources using the Single Source of Truth framework.

Pre-Post Change State Validation

Quickly determine when the operational state is not as intended.

Configuration Remediation

Automate the return to compliance of a non-compliant configuration.

ChatOps

Integrate automation to send outbound only chatbot requests. This allows for direct linking to automation system

Greenfield Site Builder

Simplify and automate new site deployments and tech refreshes codifying standard designs and integrating with your SoT.

Configuration Compliance

Quickly determine where the configuration on the device deviates from golden configurations.

Device Lifecycle Management

Gather data and generate reports about device out of maintenance or running old/unsecure software.

Circuit Maintenance Management

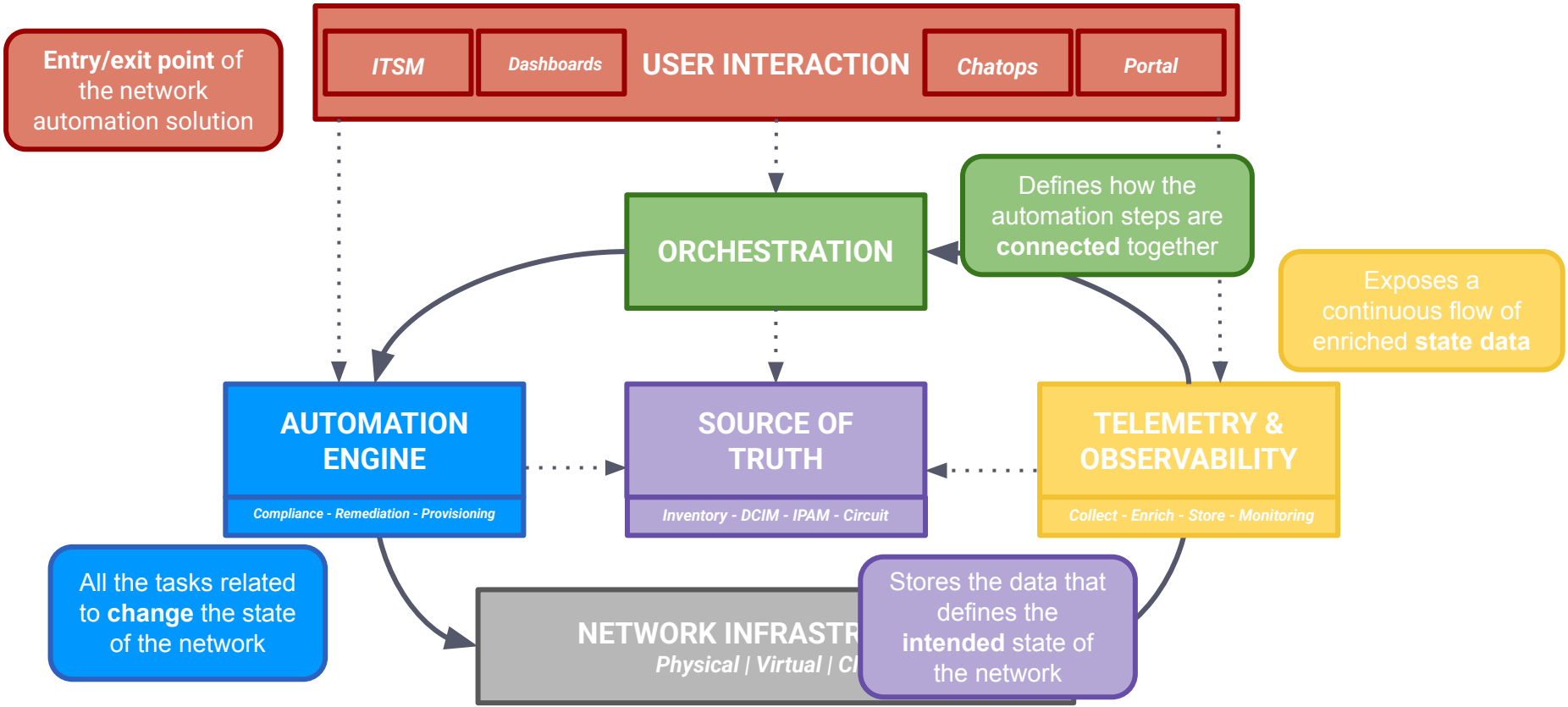
Manage circuit data and maintenance notifications dynamically in SoT.

Telemetry Analytics

Extend visibility with real time data from multiple sources and enriched with metadata

Insert YOUR own solution here

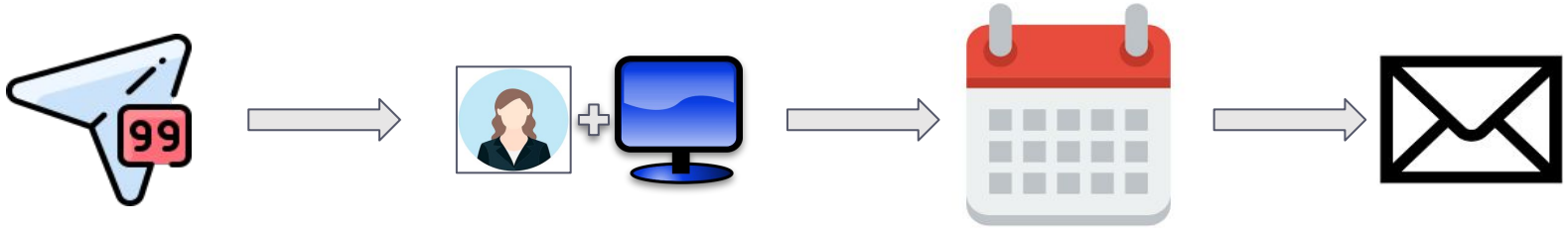
>>> Architecture



Example

How to manage Circuit Maintenances, like a pro

>>> Manual input processing of a maintenance notification



NSP sends maintenance notification

1

Network engineer parses notification and manually updates systems

2

Network engineer enters maintenance info on a calendar

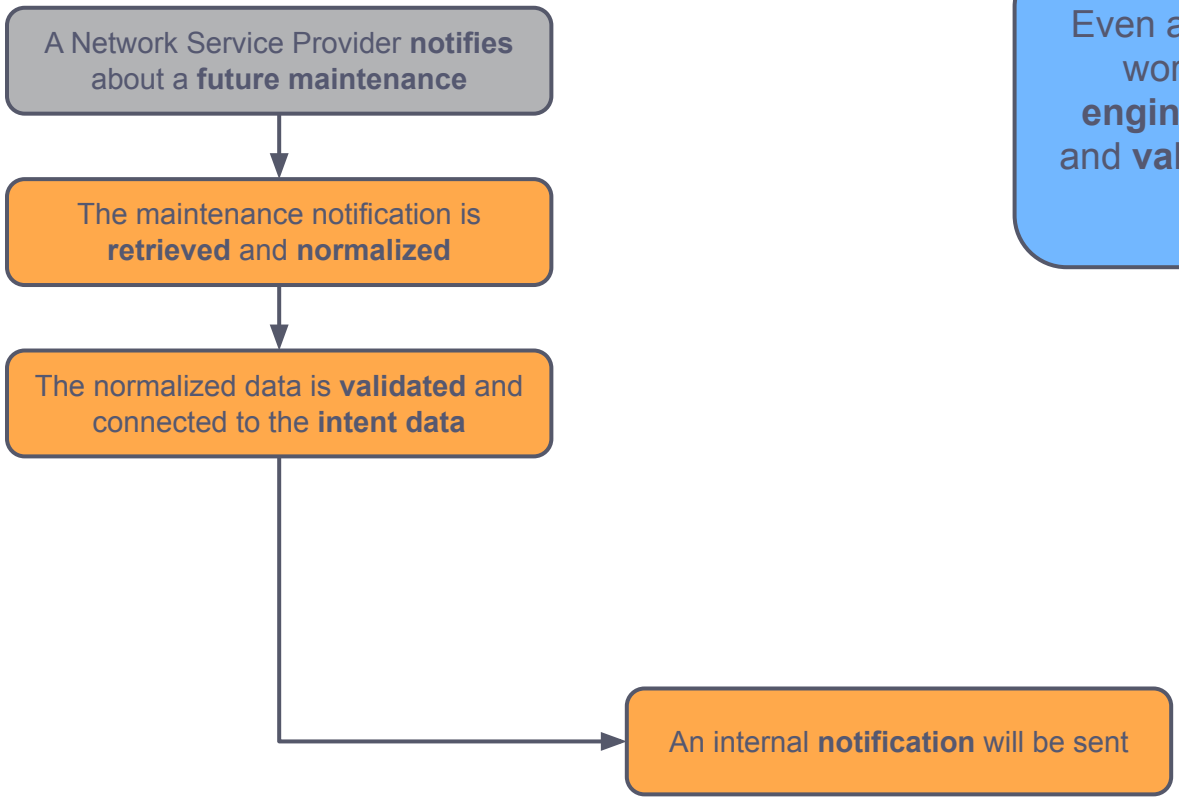
3

Network engineer emails internal group to notify of maintenance

4

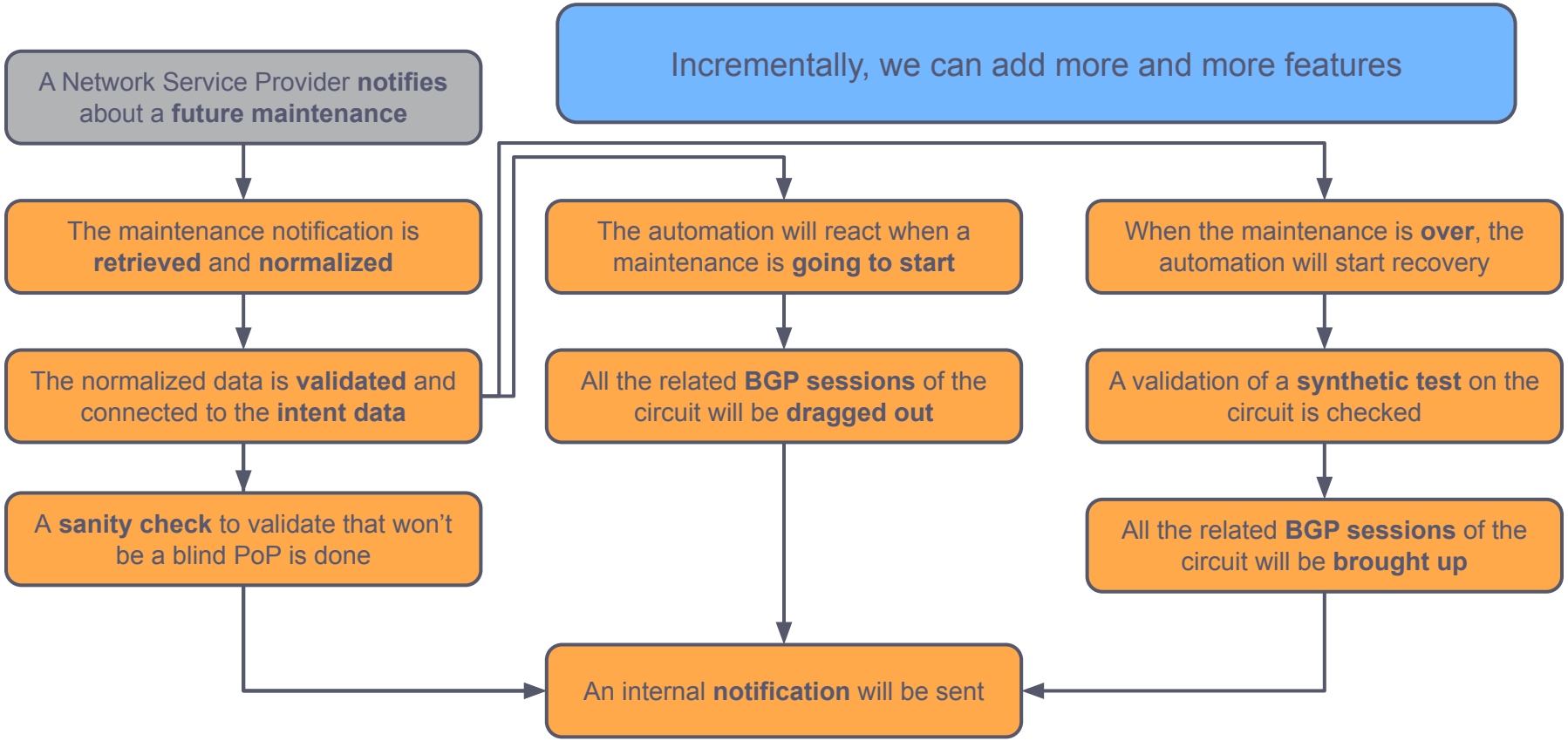
5 Maintenance window updates

>>> An Automated **Workflow**

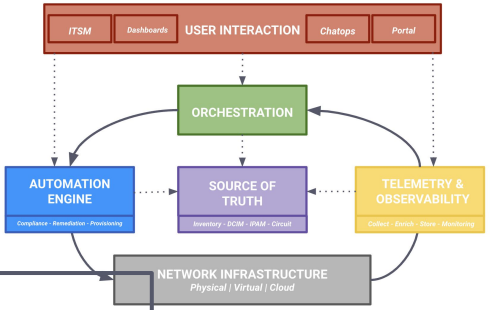


Even automating the very same manual workflow adds value: almost **zero engineering time**, data is **normalized** and **validated**, and **conveniently stored** for future consumption

>>> An Automated **Workflow**, extended



>>> Mapping to the **Architecture**



A Network Service Provider **notifies** about a **future maintenance**

SOURCE OF TRUTH

The maintenance notification is **retrieved** and **normalized**

The normalized data is **validated** and connected to the **intent data**

A **sanity check** to validate that won't be a blind PoP is done

ORCHESTRATION

The automation will react when a maintenance is **going to start**

When the maintenance is **over**, the automation will start recovery

AUTOMATION ENGINE

All the related **BGP sessions** of the circuit will be **dragged out**

All the related **BGP sessions** of the circuit will be **brought up**

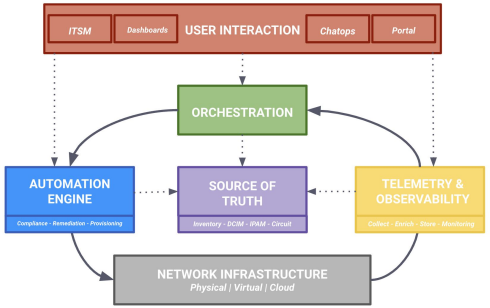
TELEMETRY & OBSERVABILITY

A validation of a **synthetic test** on the circuit is checked

USER INTERACTION

An internal **notification** will be sent

>>> Mapping to *open source* **Tools**



SOURCE OF TRUTH

git

>>> nautobot

ORCHESTRATION

AWX

AUTOMATION ENGINE

ANSIBLE

TELEMETRY & ANALYTICS

telegraf

Prometheus

USER INTERACTION

Mattermost

Grafana

Design-Driven Automation

From Architecture to Service Provisioning

WHY?

Does interacting with your network spark joy?

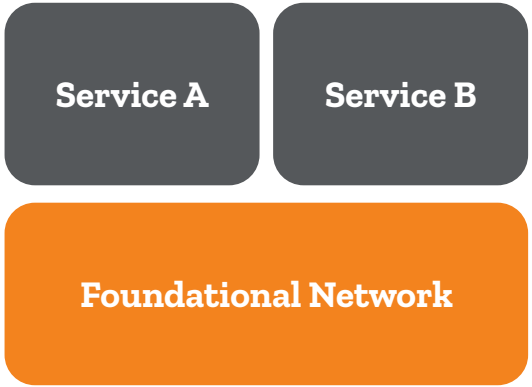
Do your users know what the network offers - and how to get it?

From the drawing board to delivery - what gets lost in translation?

Can we automate with confidence when the data is disconnected—and maintaining it feels like a full-time job?

WHAT?

Could you turn your Network into a Product?



Simplify or Feel the Pain

Data that stays connected, versioned, protected – across its entire lifecycle

Branch Site

L3VPN

New Access Switch

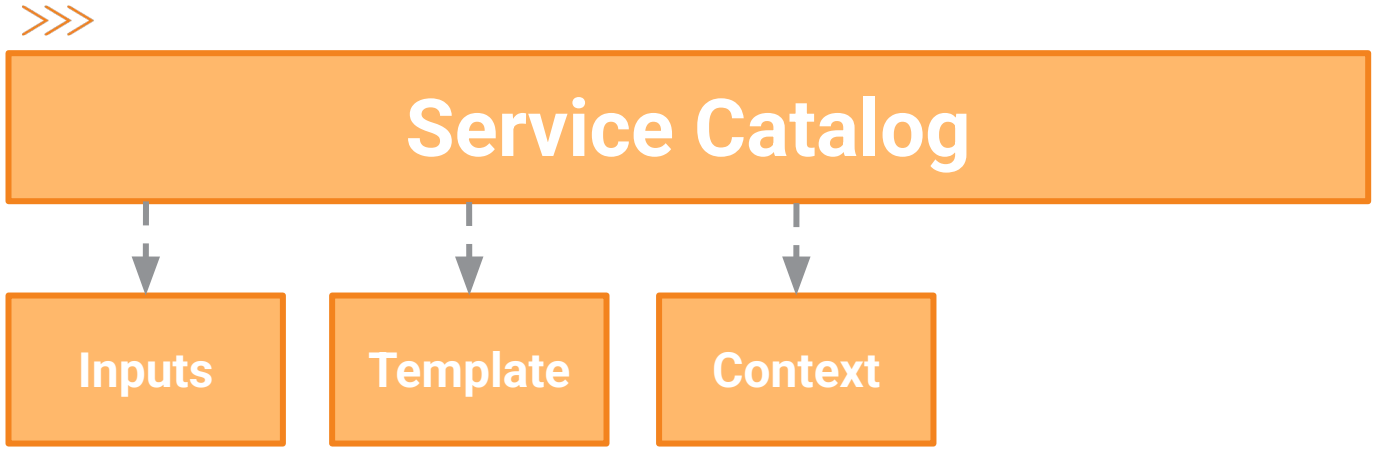
E2E Firewall Access

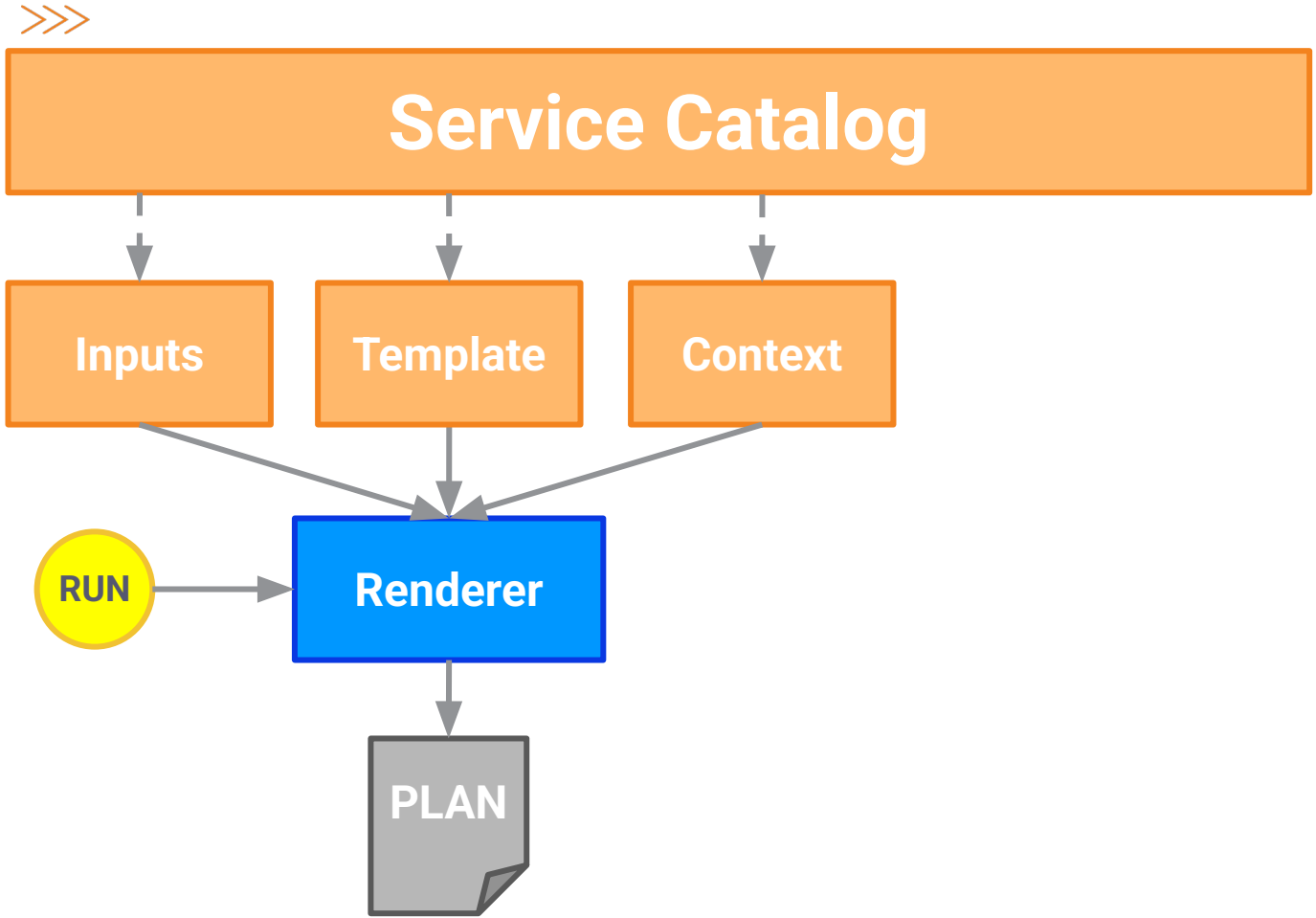
Hybrid VPN tunnel

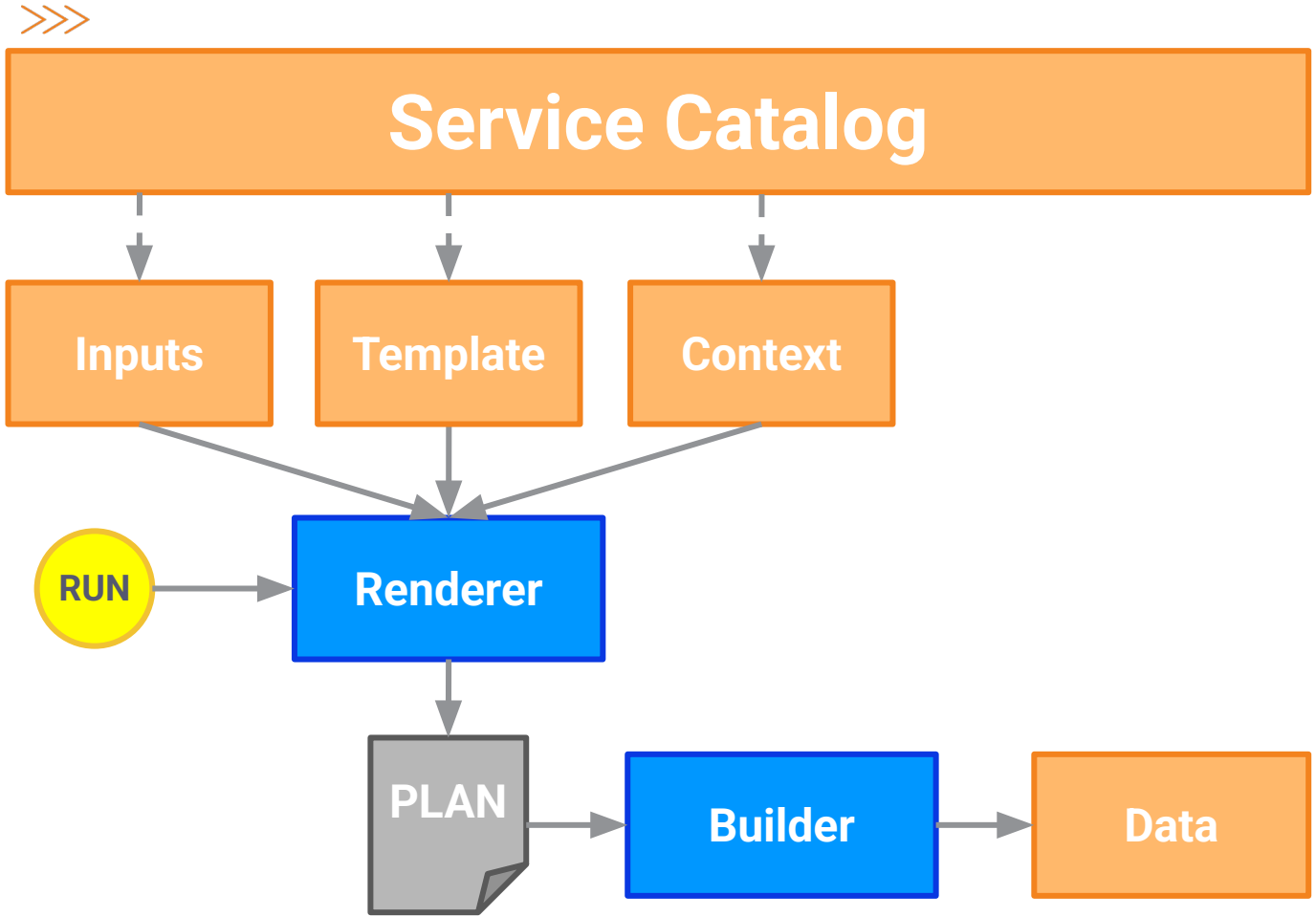
New Data Center

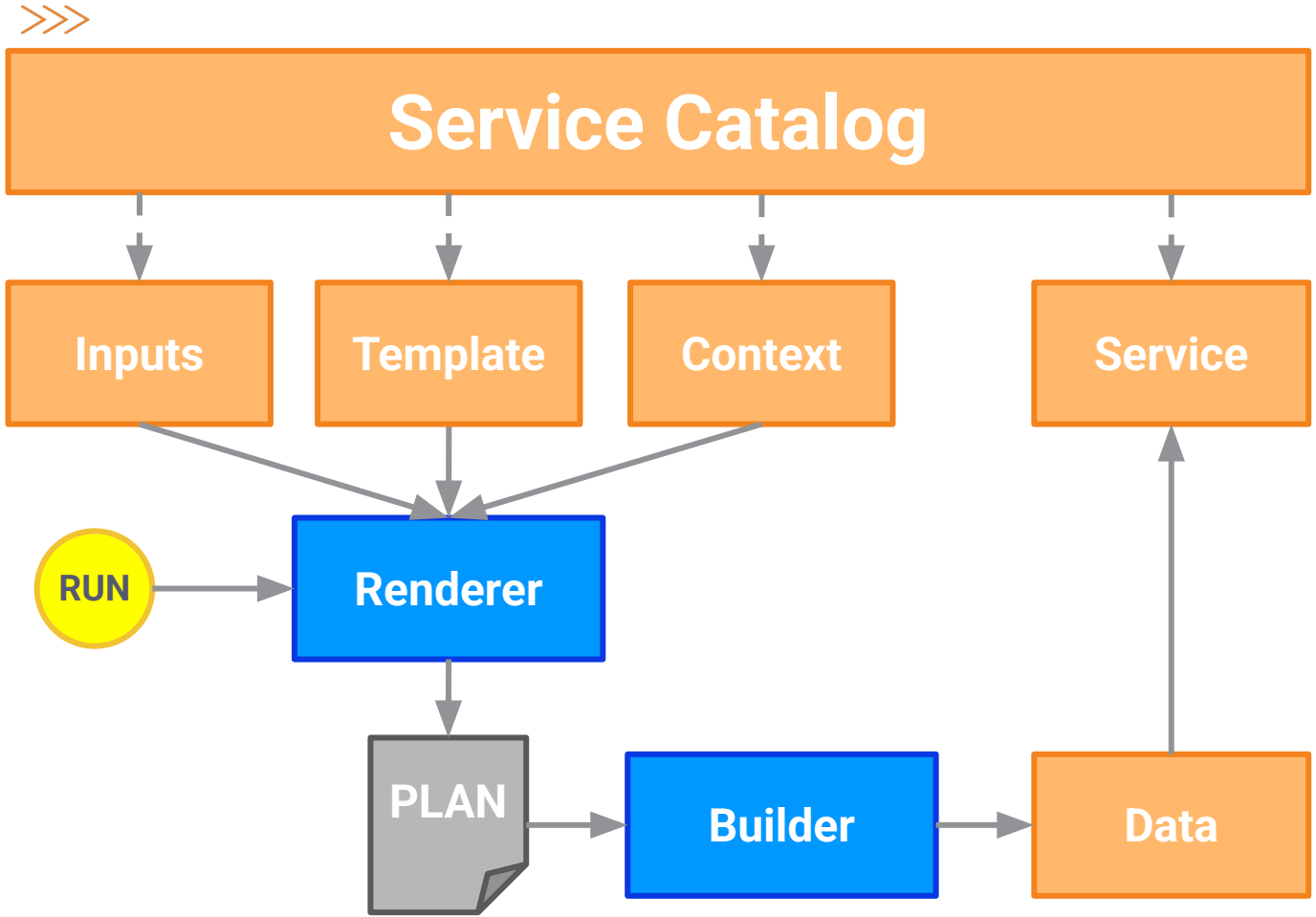
HOW?

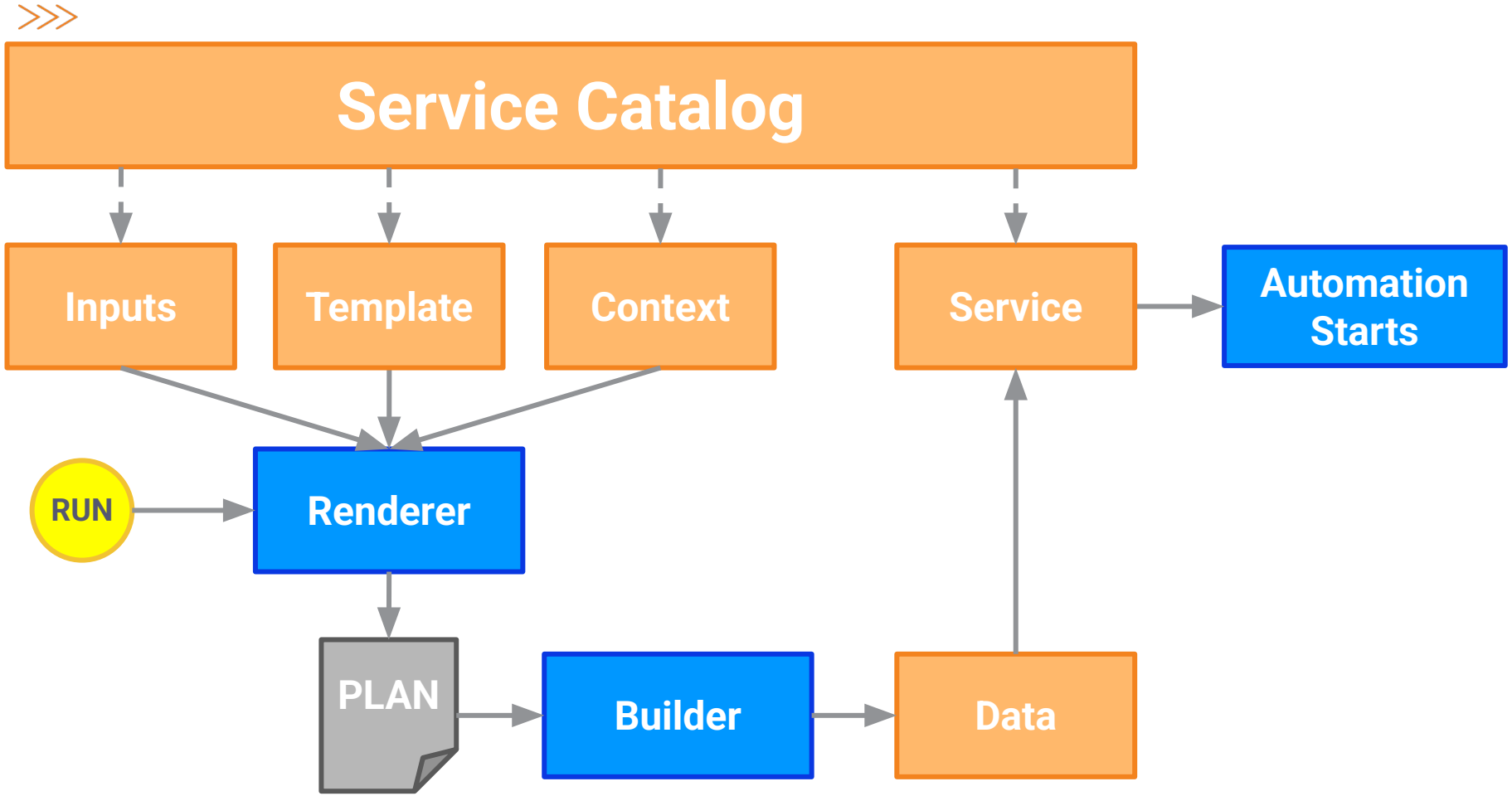
Service Catalog


Service Catalog
Inputs
Template
Context

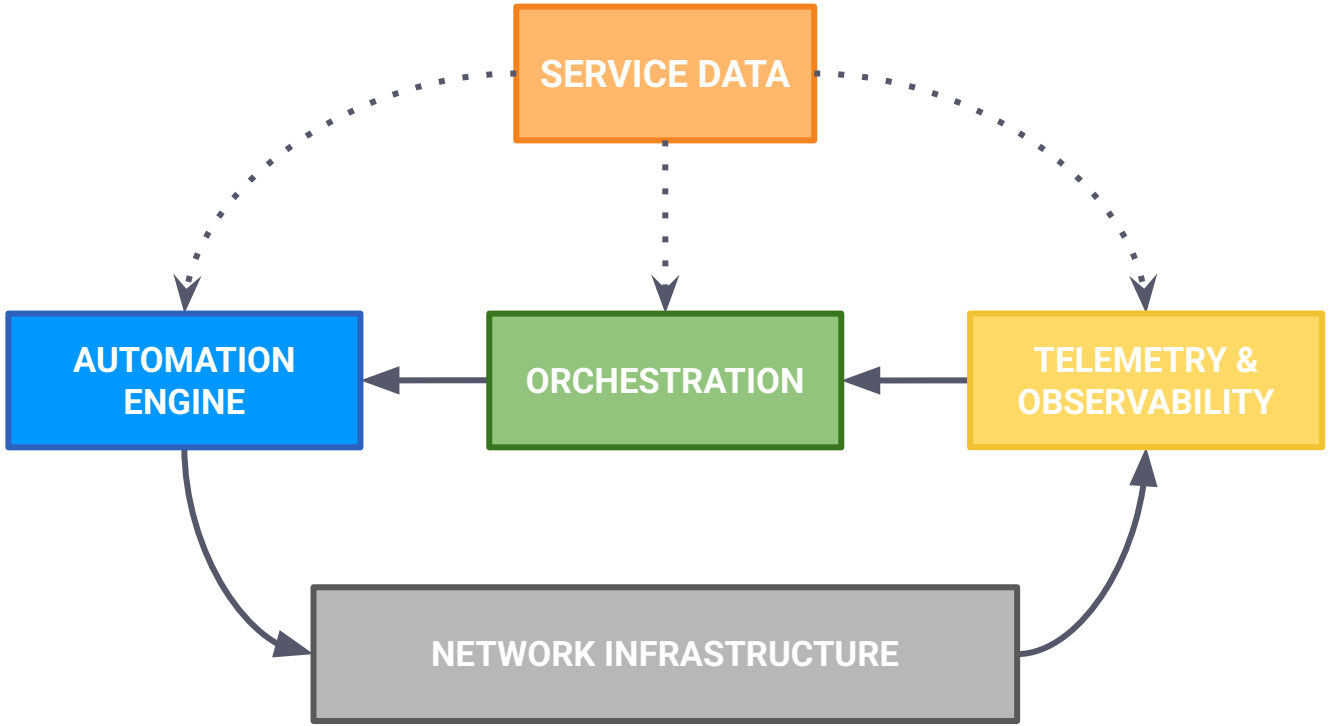

Service Catalog
Inputs
Template
Context
RUN
Renderer
PLAN


Service Catalog
Inputs
Template
Context
RUN
Renderer
PLAN
Builder
Data


Service Catalog
Inputs
Template
Context
Service
RUN
Renderer
PLAN
Builder
Data


Service Catalog
Inputs
Template
Context
Service
Automation Starts
RUN
Renderer
PLAN
Builder
Data

>>> Architecture -> Service Provisioning



Start Small

The power of compound interest

Focus on Testing the Design

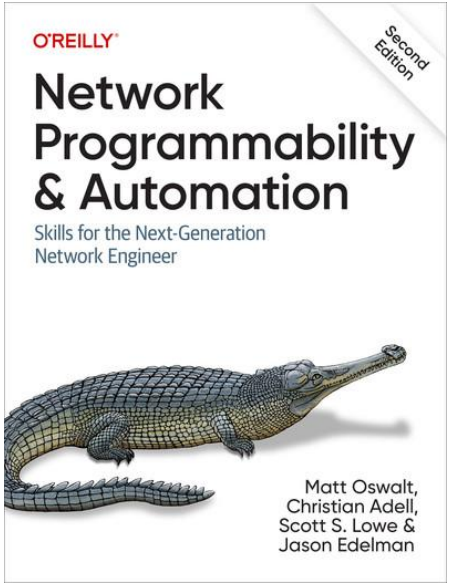
Break your network into a collection of modular services

Not just a theory

Demo Time!

>>> Learn More - Network Automation Architecture

- **Network to Code Blog**: https://blog.networktocode.com/
  - Network Automation Architecture posts: https://networktocode.com/blog/?tag=automation-architecture
- The **Network Programmability and Automation Book, 2nd Edition**
  - Chapter 14. Network Automation Architecture



More on Circuit Maintenance Automation

- **NANOG86** (Oct 17)
  - **Josh VanDeraa** session about Circuit Management automation
- **Network Field Day 27** - Automate Circuit Maintenances
  - https://www.youtube.com/watch?v=Wt_vxdGq_cs
- **RIPE 82** - How to Handle your Circuits Maintenances Proactively
  - https://ripe82.ripe.net/archives/video/516/

>>> Learn More - Design-Driven Automation

- Embracing Your Network as a Product
- Design-Driven Network Source of Truth blog
- Nautobot Design Builder App
- Autocon3 Test-Driven Design Workshop by Nikos Kallergis
- Nanog 88 - Design Driven Network Assurance by Jeremy Schulman
- Nanog 93 - Now you can too! Design-driven automation by Wim Van Deun

>>> Example Tools References

- Nautobot
  - https://github.com/nautobot/nautobot
- Nautobot Circuit Maintenance Plugin
  - https://github.com/nautobot/nautobot-plugin-circuit-maintenance
- Circuit Maintenance Parser
  - https://github.com/networktocode/circuit-maintenance-parser
- AWX
  - https://github.com/ansible/awx
- Ansible
  - https://github.com/ansible/ansible
- Telegraf
  - https://github.com/influxdata/telegraf
- Prometheus
  - https://github.com/prometheus/prometheus
- Grafana
  - https://github.com/grafana/grafana
- Mattermost
  - https://github.com/mattermost

>>> network .toCode()

Labai Ačiū!

ttl255.com

przemek-rogala