

Network Automation Big Picture overview 2023

A quick look at what's going on

Steinn Örvar Bjarnarson 2023

Who is this guy?



- **Steinn Orvar Bjarnarson (Steinzi)**
- Lead product development engineer at Advania Iceland
- Network engineer turned Automation specialist/developer
 - Cisco CCIE #60715
 - Fortinet NSE7
 - Juniper Specialist
- BSc Computer Science



Overview

Outline of the talk

Intro

Automation Big Picture

End-to-end “Lego” Orchestration

Imperative vs Declarative

Network automation tier list

Tools, tips, and tricks!

Conclusion

Q&A

What is the goal of network automation?

Answer: It depends...

Automation Goals

It depends, but here are some of mine

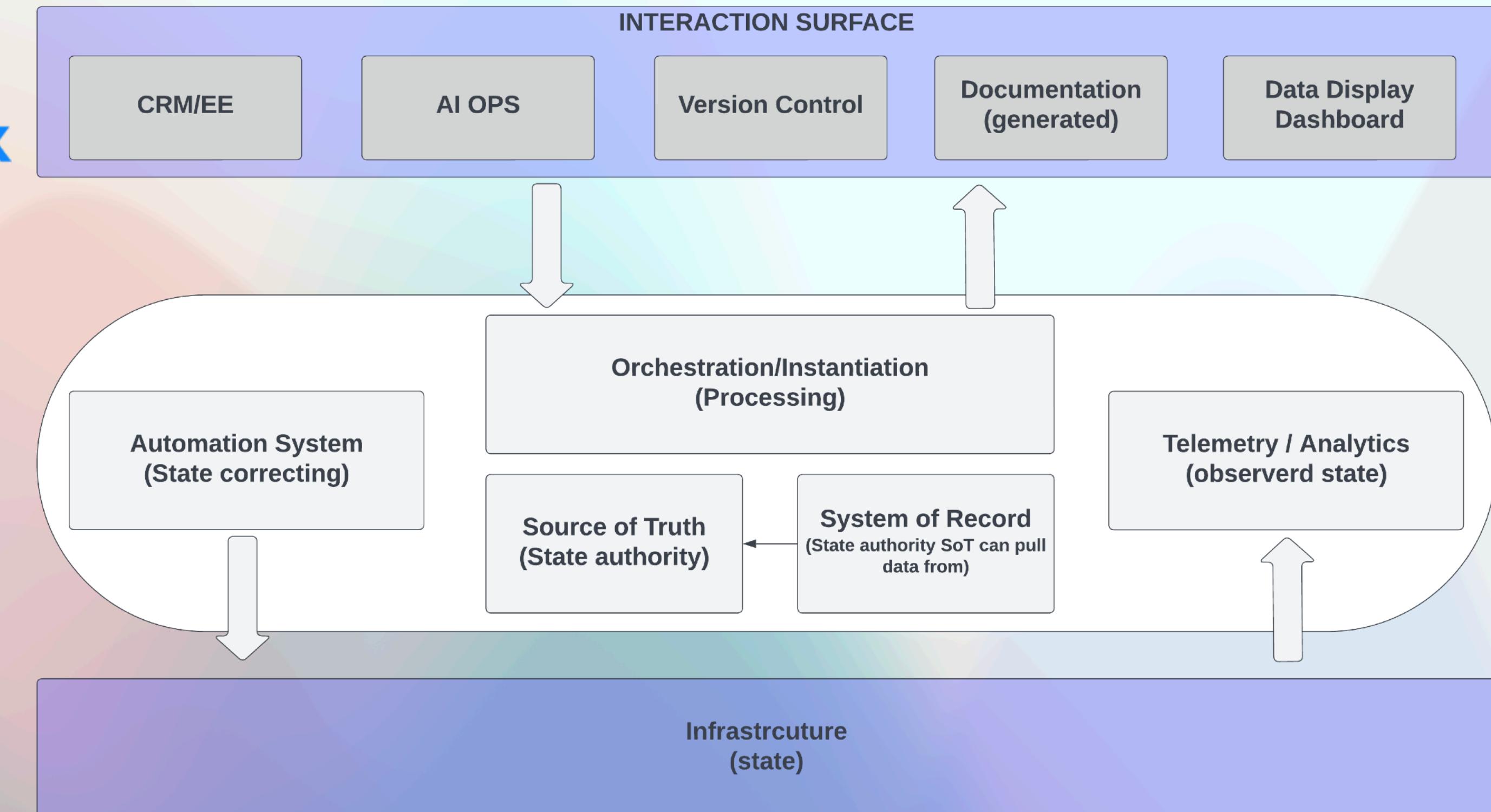
- 0 Concise definition of services
- 1 Consistent delivery of service to customers
- 2 Automatic deployment validation CI/CD style
- 3 Automatic Rollback of any deployment with errors.
- 4 End to End service definitions
- 5 Faster turnaround time on tasks & tickets
- 6 Customer self-service infrastructure
- 7 Telemetry and debug data collection and analysis, (proactive not reactive)

Network Automation: Big Picture and Basic Components

Network Automation

Big-picture

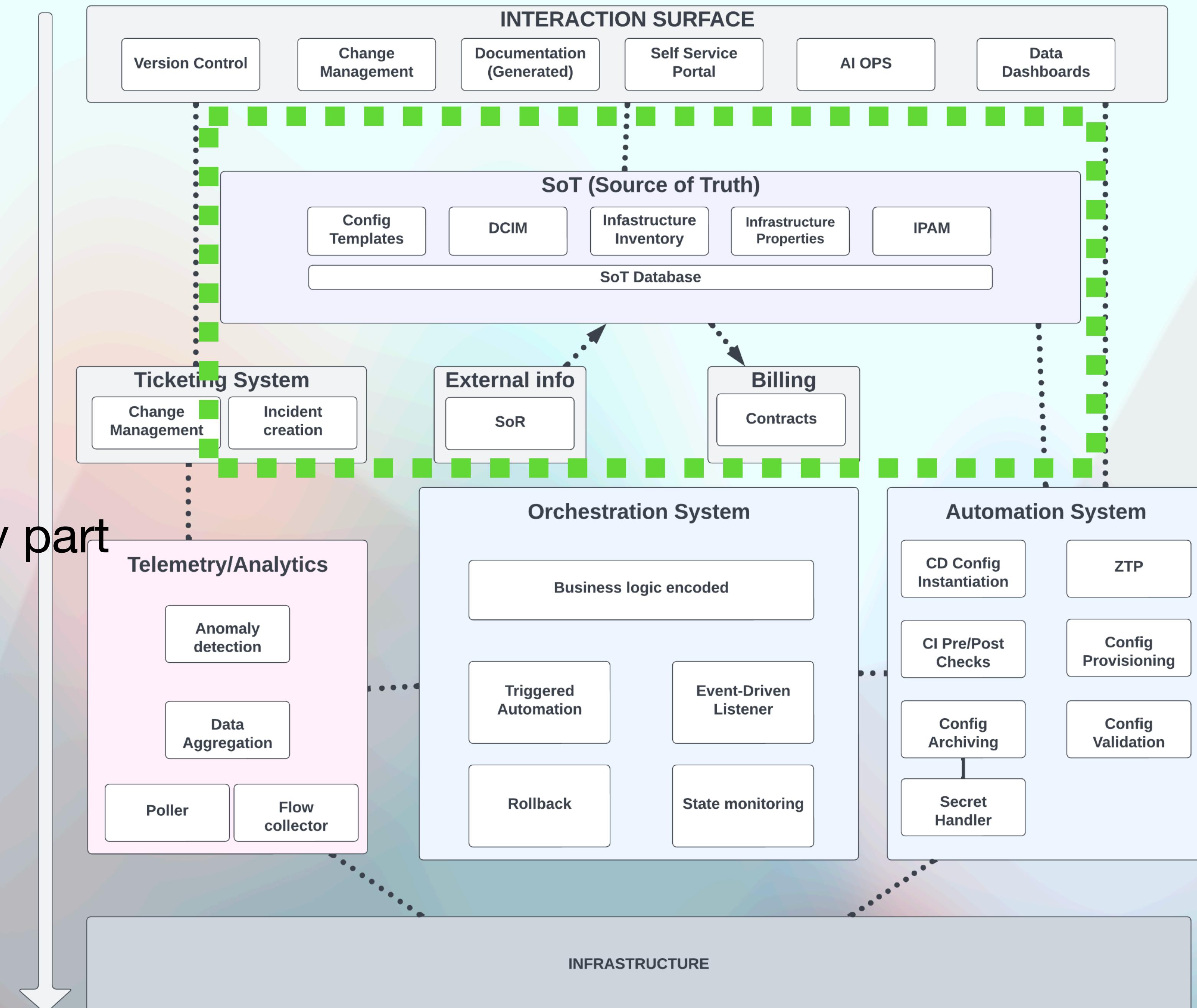
- Source of truth (SoT)
- System of record (SoR)
- Orchestration/Automation system
- Telemetry & Monitoring
- User interaction



Source of Truth

Heart of the system

- Holiest of holies
- Netbox/Nautobot/DB
- SoR integration is the tricky part
- What is an SoR?



Independent Source of Truth

Vendor lockin

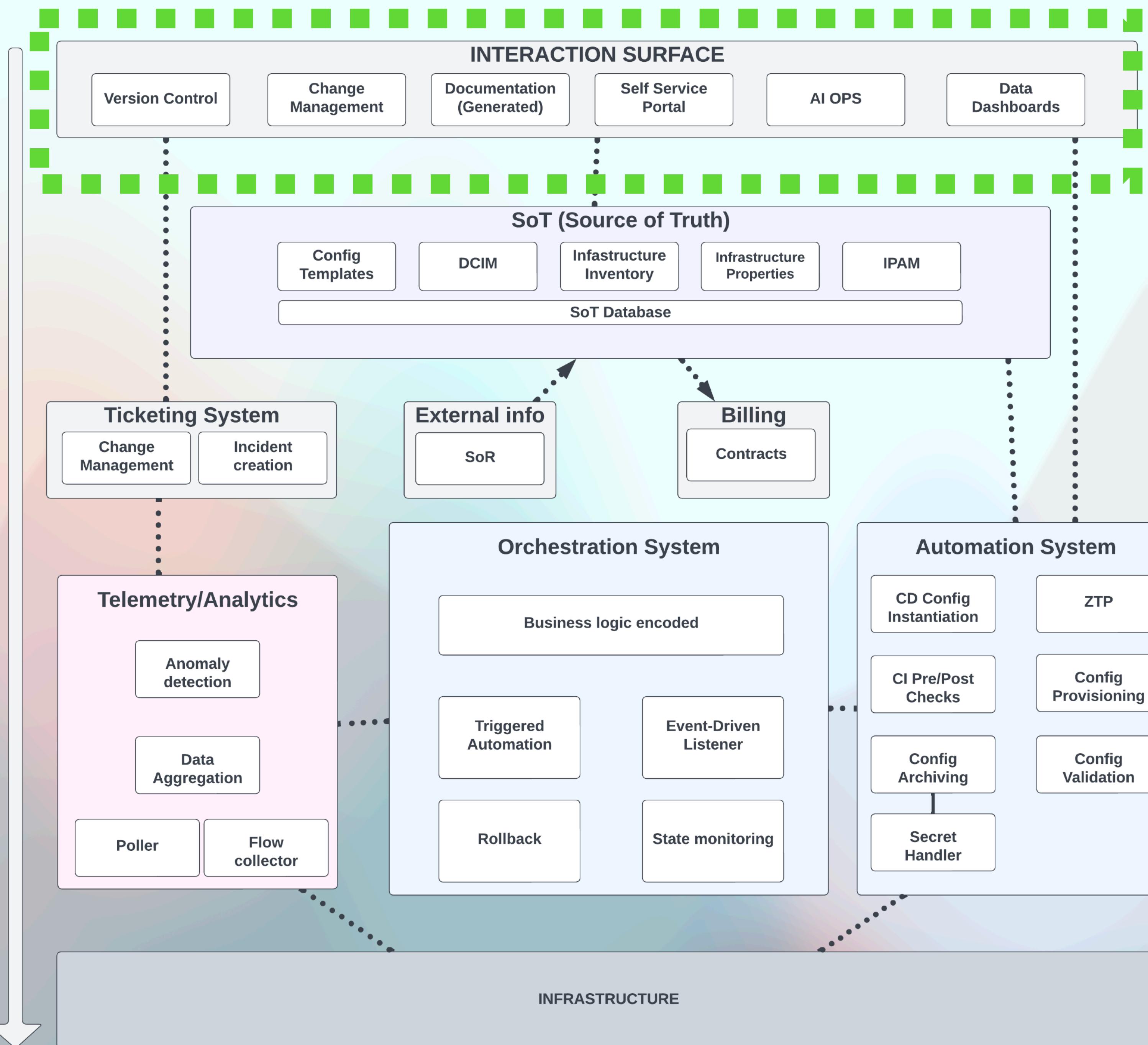
- You might want to avoid it...
- Decoupling automation components is the key.
- API-first-driven solutions are a priority



User interaction

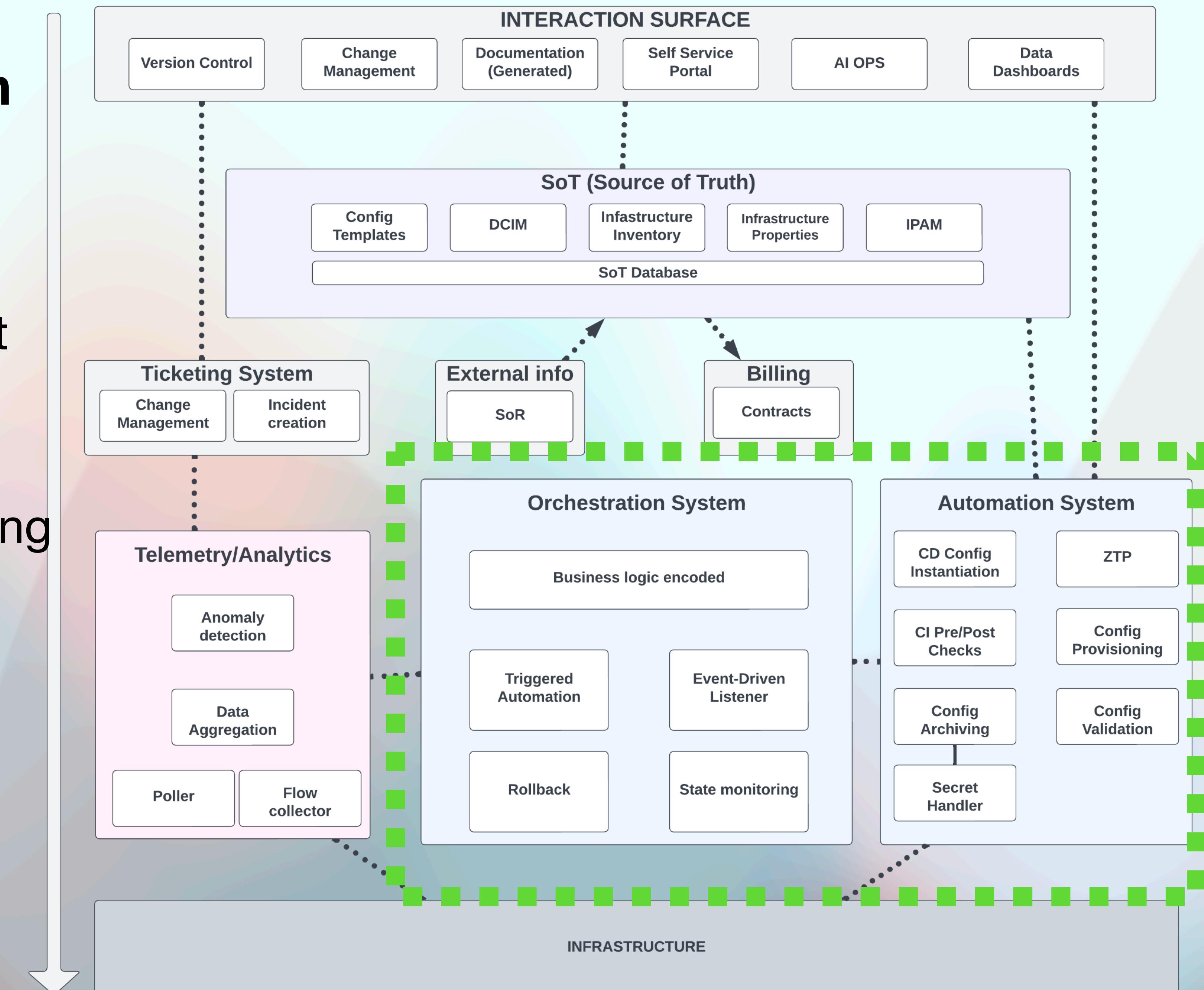
Please keep your hands off the network

- Abstract business logic and processes into other systems
- Consumable parts like Documentation
- Good change management is critical!



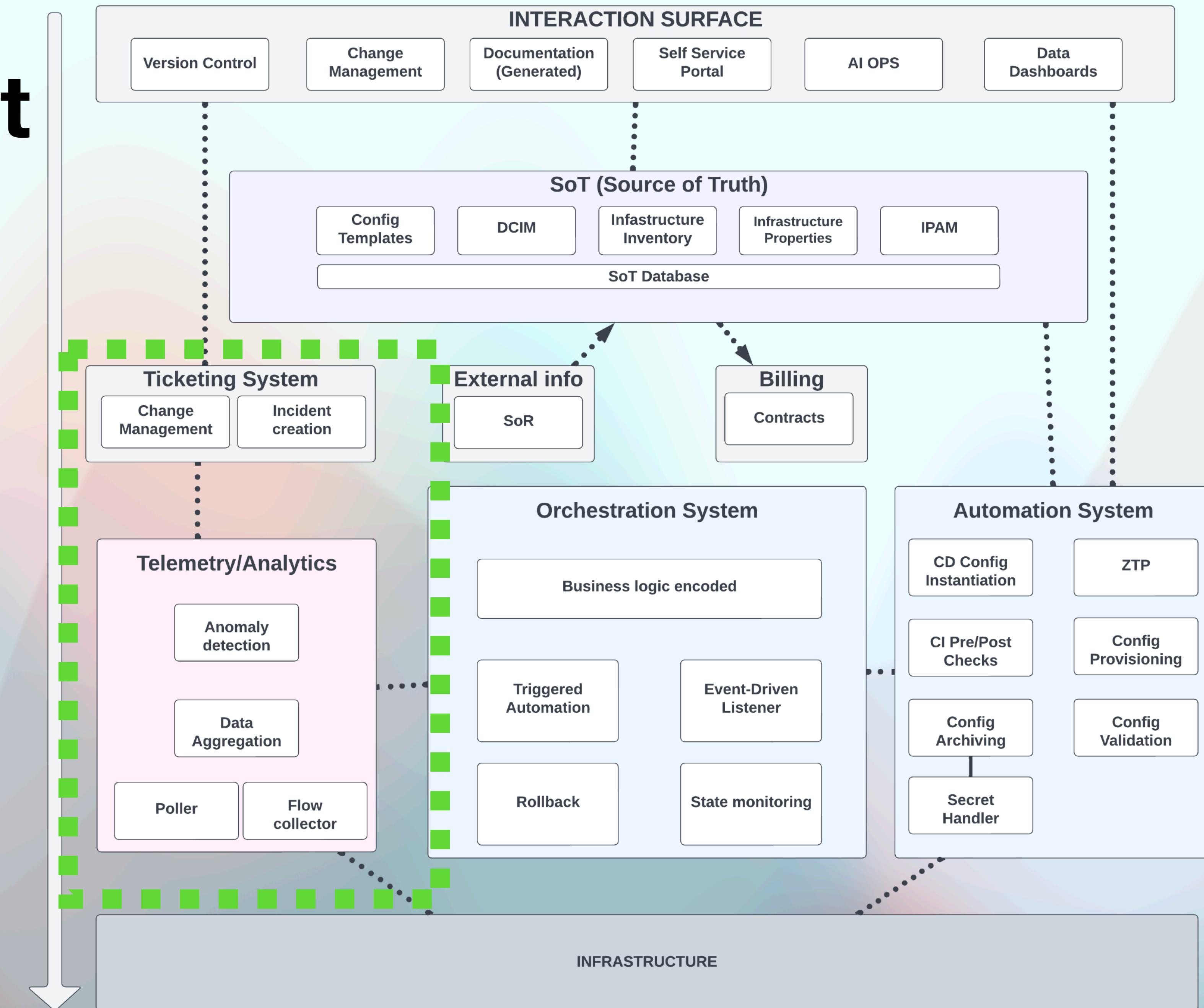
Automation/Orchestration where the magic happens

- Lots of options for this part
- More on this later!
- Watch out for vendors selling you solutions that handle everything here!



Telemetry/Ticket Alert! Alert!

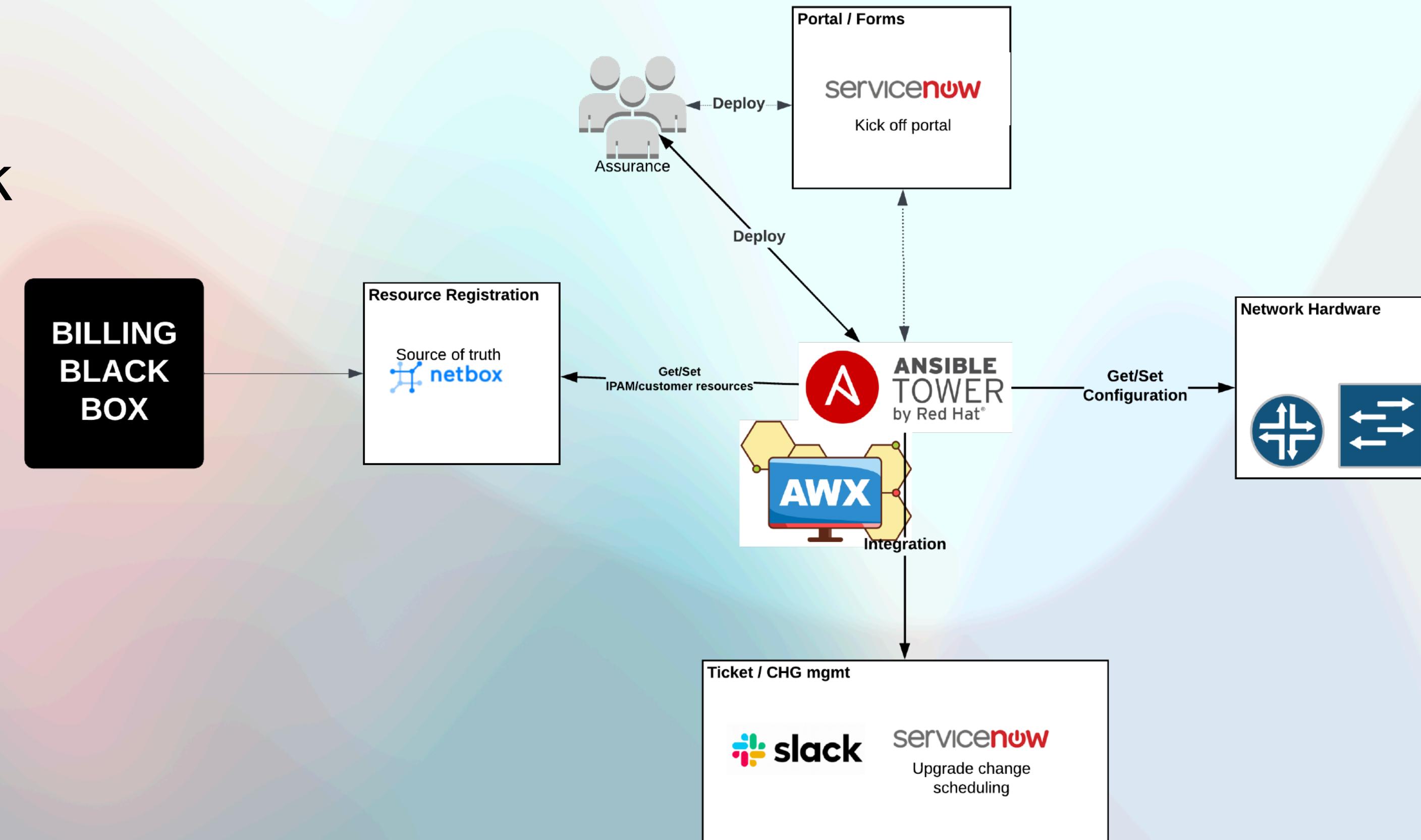
- Incident creation
- Monitoring of our systems



Why go through all this?

Benefits of this design

- Separation of billing and network vendors & services
- Authoritative state source (SoT)
- Multivendor support
- Sensible deployment defaults
- Reduced human error
- End 2 end service definitions?

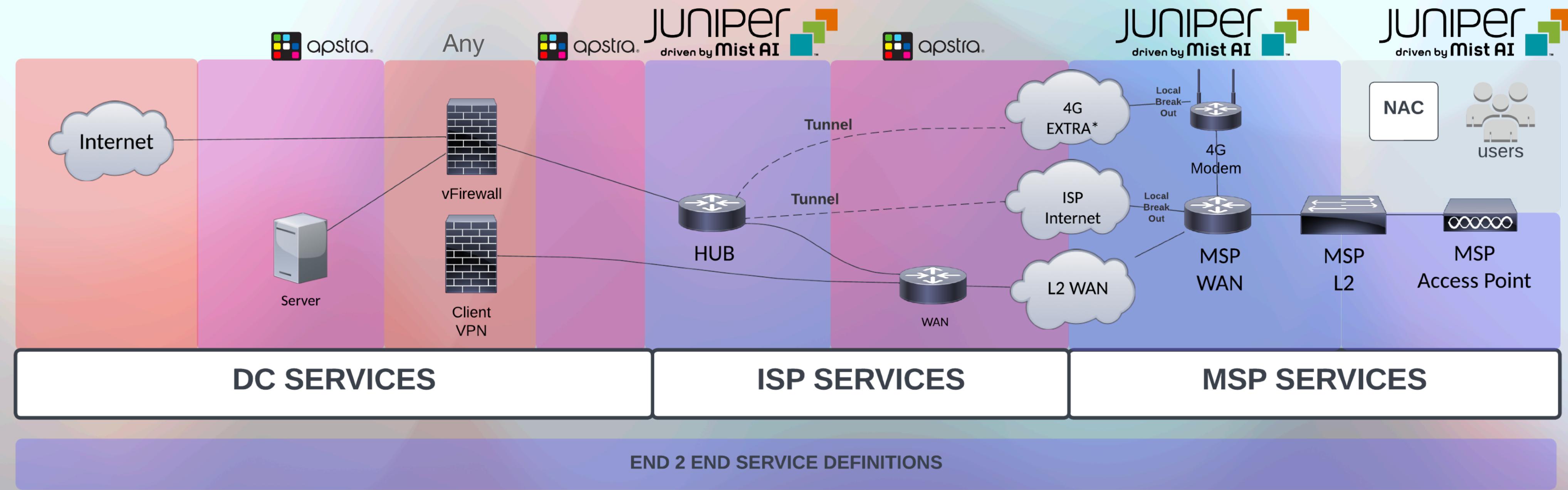
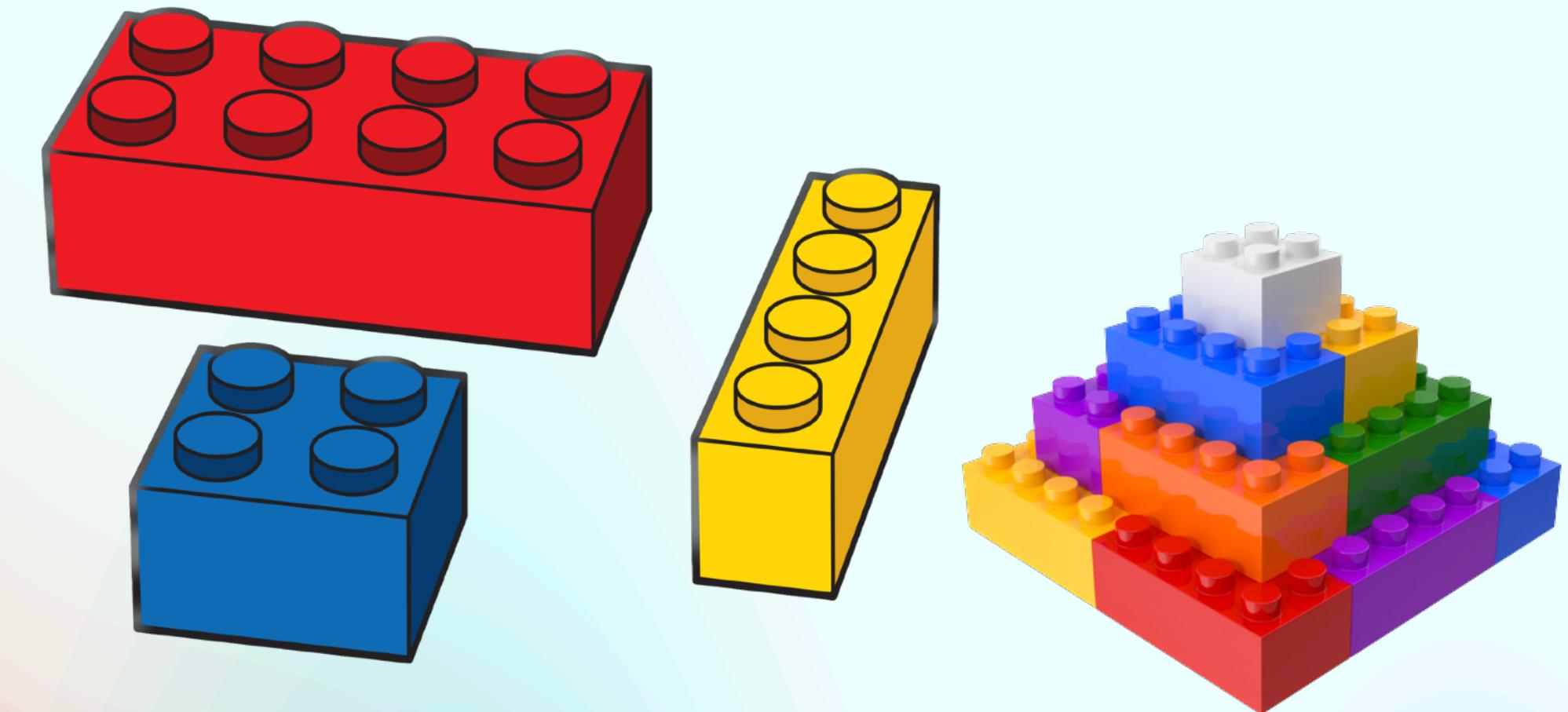


End-to-end Service Chaining and Orchestration

Network Service Orchestration

“Lego orchestration”

- End 2 end service chaining/definitions
- Service defined from DC/Cloud to the user
- Billing and resource mapping to customers



Let's have Pasta

Imperative Step-by-step (HOW)

VS.

Declarative: The desired result (WHAT)

Imperative (HOW) Step-by-step Recipe

- We focus on the **HOW**, and we should get **WHAT** we want
- The recipe lists each step in detail
- Highly coupled in most cases
- We get pasta



Klaus Nielsen

```
begin
    fill_pot_with_water()
    place_pot_on_stove()
    turn_stove_on("high")
    wait_for_water_to_boil()

    add_salt_to_boiling_water()
    open_pasta_box()
    add_pasta_to_boiling_water("spaghetti")
    set_timer("al dente")
    stir_pasta_periodically()

    while timer_is_running:
        chop_onion()
        chop_garlic()
        heat_pan_on_stove("medium")
        add_olive_oil_to_pan()
        saute_onion_and_garlic()

        open_canned_tomatoes()
        add_tomatoes_to_pan()
        add_seasonings("basil", "oregano", "salt", "pepper")
        simmer_sauce()

        prepare_meatball_mixture()
        shape_meatballs()
        heat_another_pan_on_stove("medium")
        add_olive_oil_to_pan()
        cook_meatballs_in_pan()

    stop_timer()
    drain_pasta_in_colander()

    add_cooked_pasta_to_sauce()
    mix_pasta_and_sauce()
    add_cooked_meatballs_to_pasta()
    grate_parmesan_cheese()
    sprinkle_cheese_on_top()

    serve_pasta_on_plate()
end
```

Declarative (WHAT)

Desired end result

- We order **WHAT** we want by declaring it to the waiter.
- Focus only on outcomes.
- I don't care how you do it.
- We get pasta.
- What about the cooks in the kitchen?



[RODNAE Productions](#)

```
1  order:  
2    dish: pasta  
3    pasta_type: spaghetti  
4    sauce: marinara  
5    extras:  
6      - meatballs  
7      - parmesan_cheese  
8    al_dente: True  
9
```

Sudo code comparison

Imperative

```
begin
    fill_pot_with_water()
    place_pot_on_stove()
    turn_stove_on("high")
    wait_for_water_to_boil()

    add_salt_to_boiling_water()
    open_pasta_box()
    add_pasta_to_boiling_water("spaghetti")
    set_timer("al dente")
    stir_pasta_periodically()

    while timer_is_running:
        chop_onion()
        chop_garlic()
        heat_pan_on_stove("medium")
        add_olive_oil_to_pan()
        saute_onion_and_garlic()

        open_canned_tomatoes()
        add_tomatoes_to_pan()
        add_seasonings("basil", "oregano", "salt", "pepper")
        simmer_sauce()

        prepare_meatball_mixture()
        shape_meatballs()
        heat_another_pan_on_stove("medium")
        add_olive_oil_to_pan()
        cook_meatballs_in_pan()

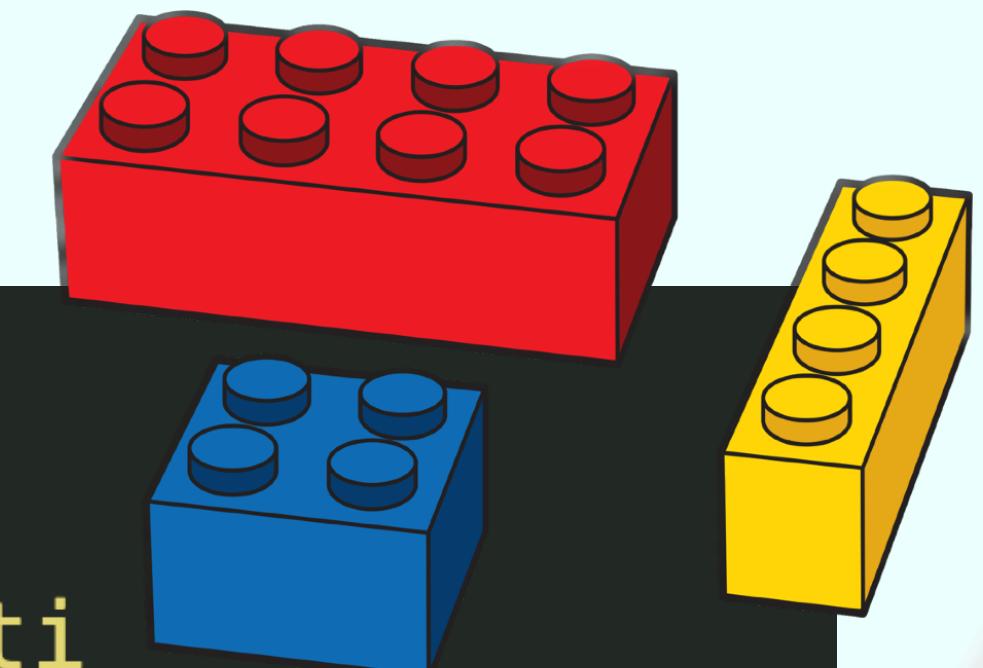
    stop_timer()
    drain_pasta_in_colander()

    add_cooked_pasta_to_sauce()
    mix_pasta_and_sauce()
    add_cooked_meatballs_to_pasta()
```



Declarative

```
1   order:
2       dish: pasta
3       pasta_type: spaghetti
4       sauce: marinara
5       extras:
6           - meatballs
7           - parmesan_cheese
8   al_dente: True
9
```



COMPLEX UNDER THE HOOD
(EASY TO USE)

Automation abstraction level

What does my automation target?

- Try to go as high as we can
- This depends on a lot of factors
- No one-size-fits-all
- Build your plan from the bottom up

Customer/Tenant

Site

Switches

Switch

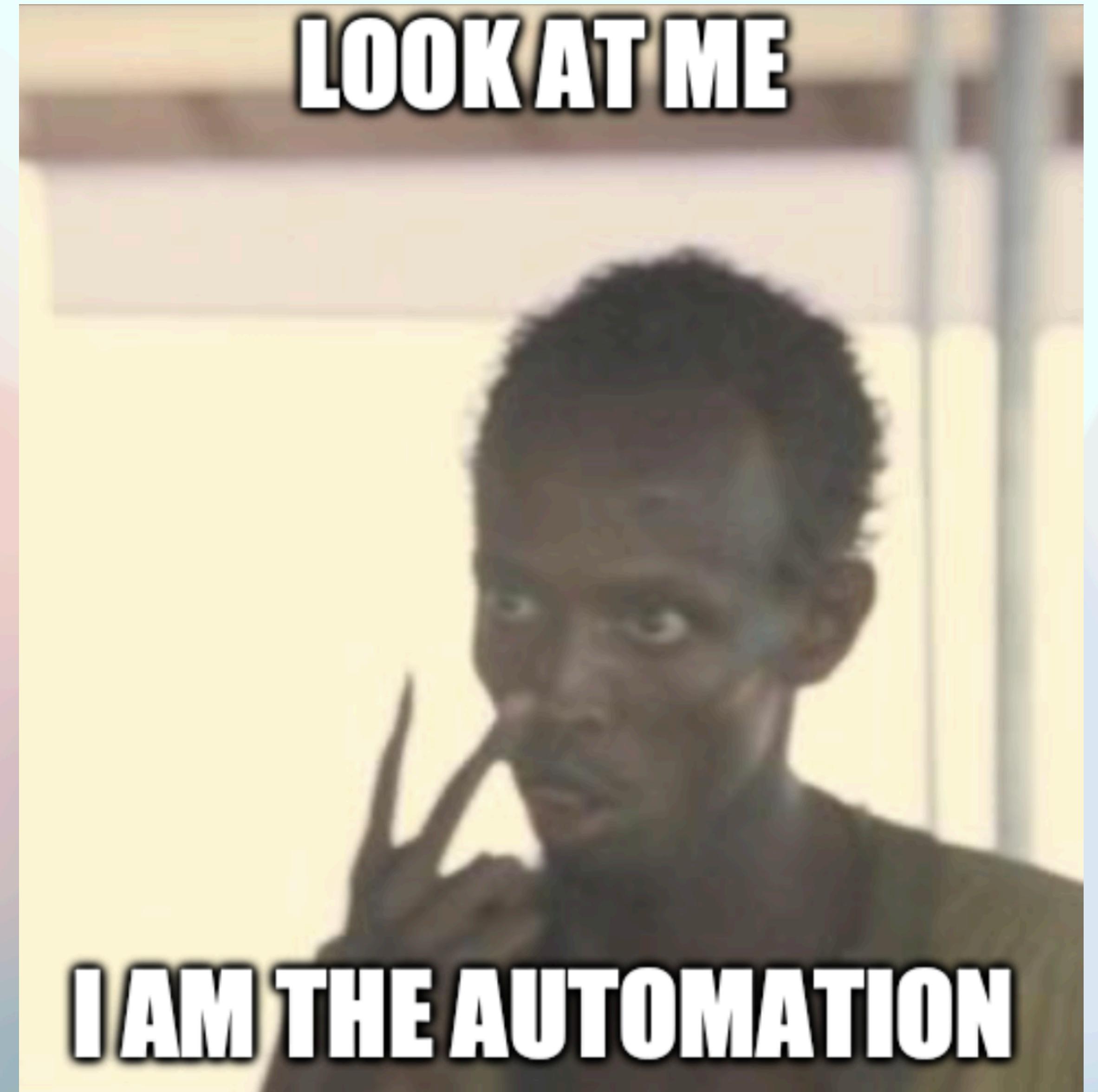
Port

Automation journey tier list!

Tier -1

Meat-based automation!

- You are the automation...
- You have a ticketing system +1
- You have a Wiki with old articles written by god knows who...+1
- A OC/NOC that customers can call in the middle of the night +1
- You have a DCIM/IPAM, not in excel... +1



Tier 0

Basics

- keeping track of manual changes
+1
- Rollback plan for change requests! +1
- Automatic backup of configuration at N time intervals
+1

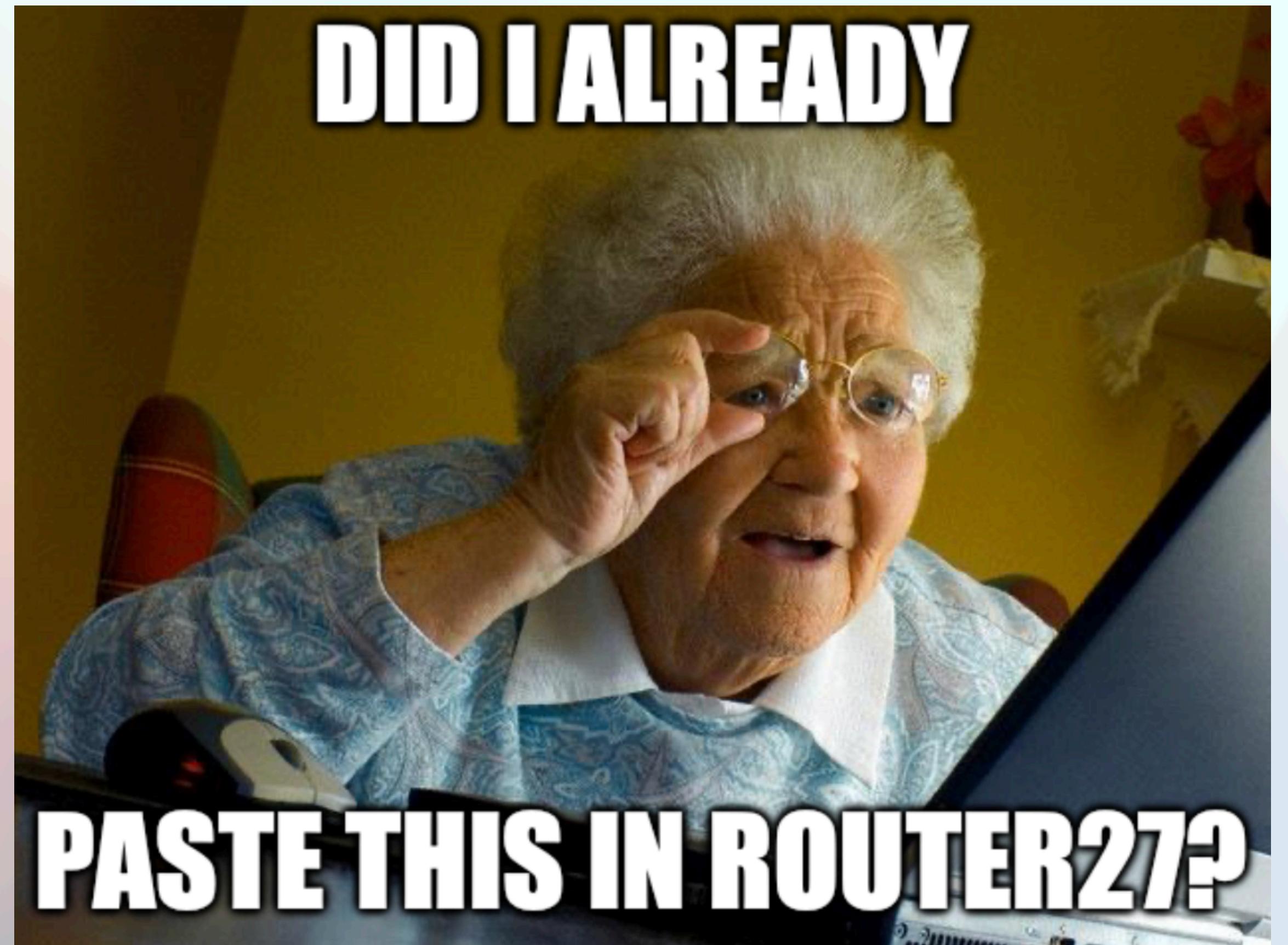
Hey do you have
a copy of the config
before the change?



Tier 1

Config templates

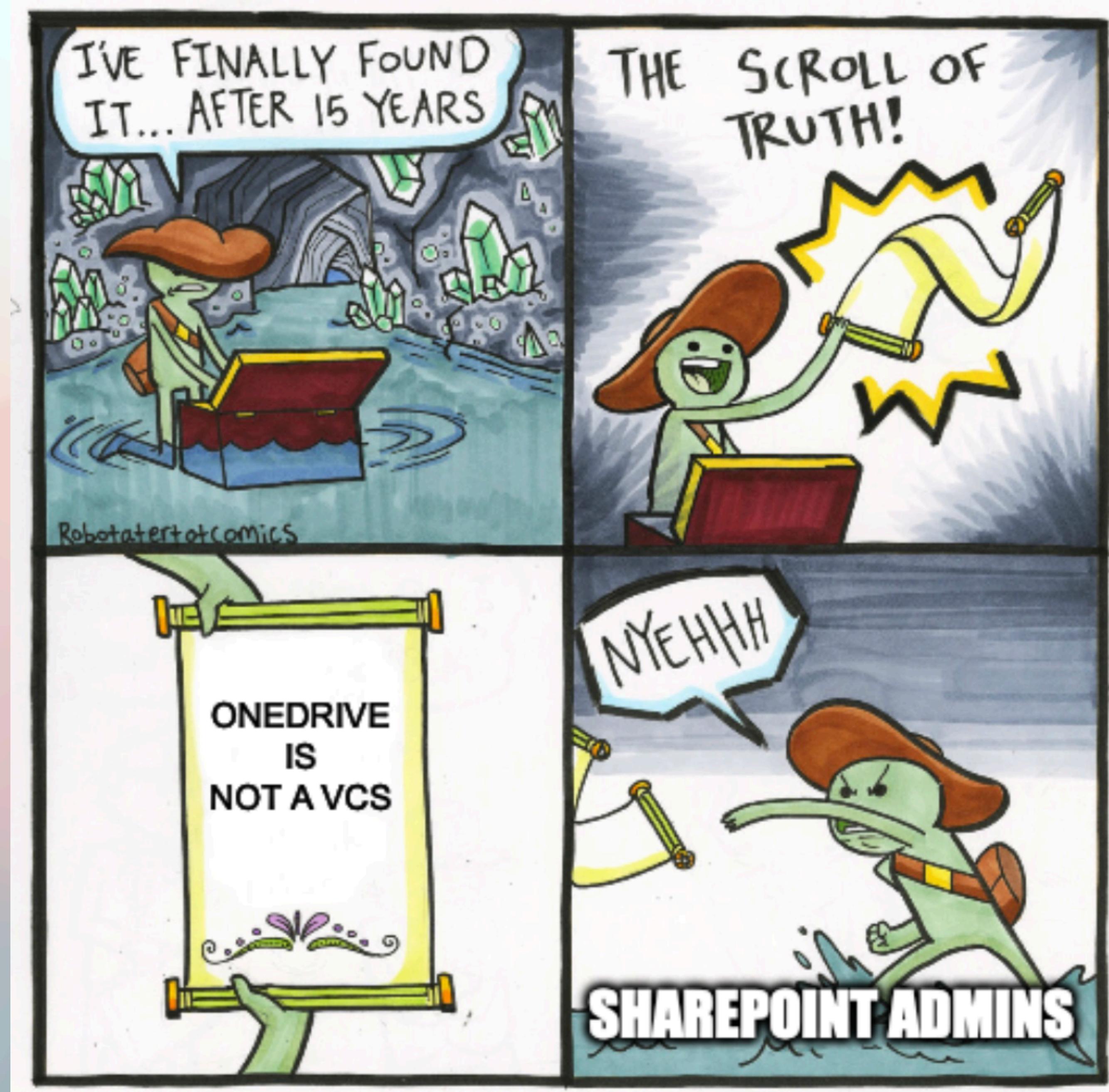
- Find and replace {{IP_ADDRESS}} in a CLI config template
- Copy and paste it into the device by hand during change windows
- OH NO! NO! I PASTED IT TWICE!



Tier 2

Laptop level

- Solo scripts that do one job well +1
- Each team member has a shell script they run on their local machine
- no audibility -1
- Hardcoded password & API keys -9999
- no rollback handling... -1
- The person quits, and well.... there go the scripts. -1



Tier 3

Centralised Automation environment

- User auth +1
- 2FA +1
- Audit trail +1
- Secured environment +1
- no hardcoded passwords/API keys +9999



Tier 4

Law and order

- Standardized around an automation framework +1 
- Most things are written in the same language +1
- You have an automation dictator or council that guide development +1



Tier 5

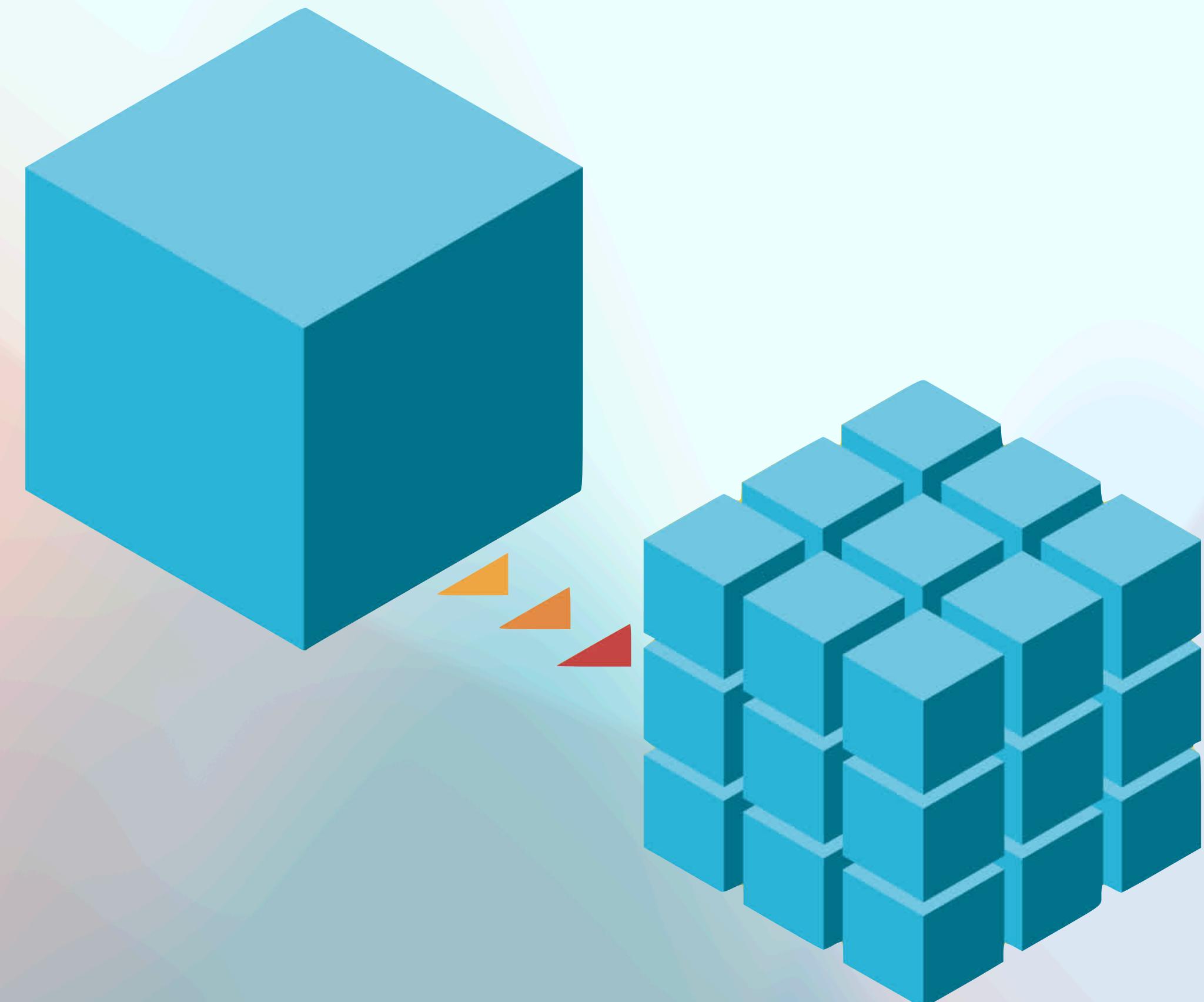
Business processes driven automation

- Customer Relationship Management (CRM)
- Enterprise Service Management (ESM)
- Connected and driven workflows
- Customer lifecycle management
 - On-board to off-board!

Tier 6

More Microservices!

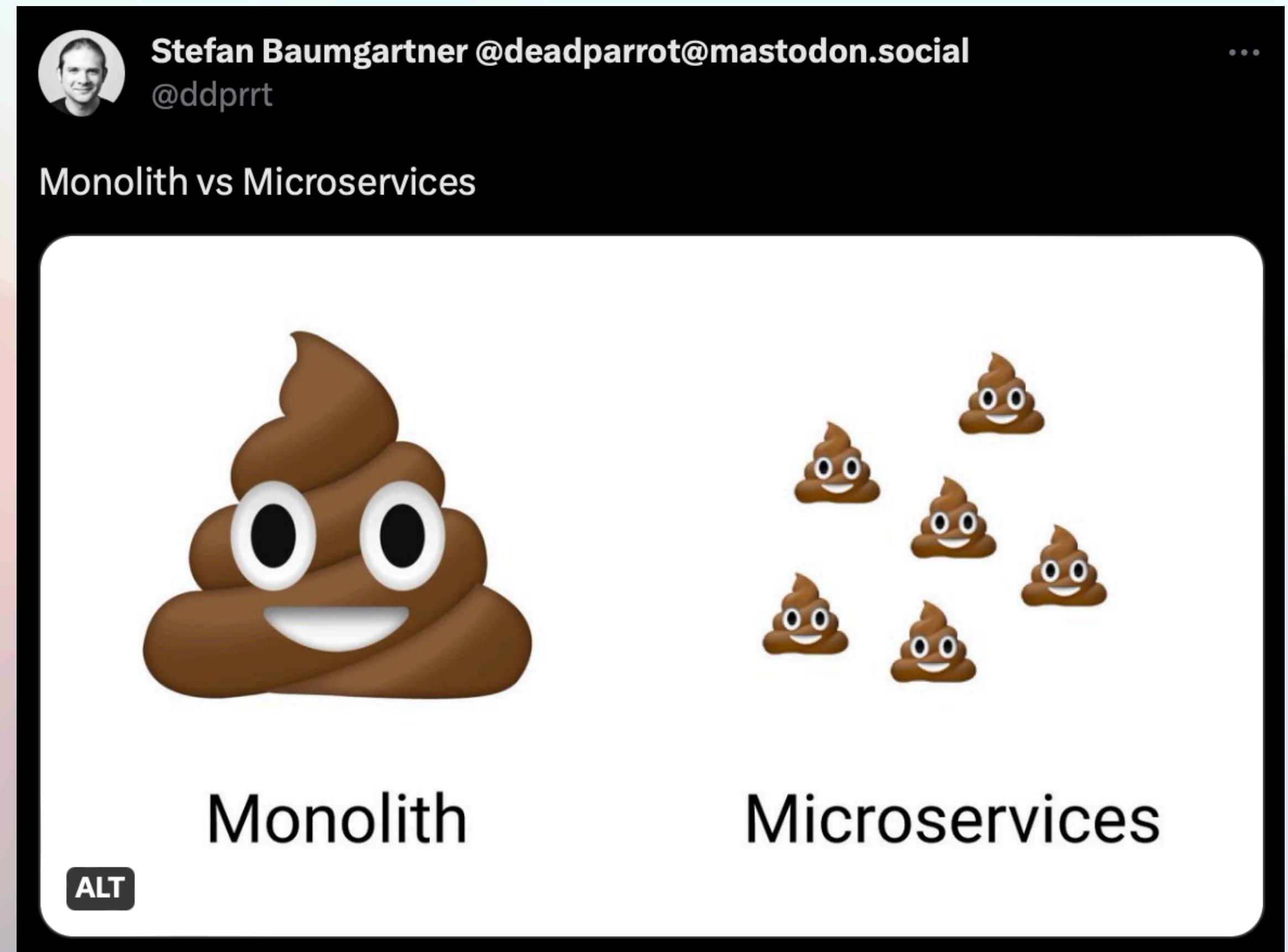
- We start to break everything into Microservices +1
- Creating boundaries between business functions and services +1



Tier 7

Less Microservices!

- We find out that maintaining all these Microservices is insane +1
- We eliminate stuff we don't need and have no idea how it got into production. +1



Tier 8

Event-driven microservice architecture

- The event-driven people show up
- Start talking about CQRS and Sagas
- Someone says something about RabbitMQ and Kafka
- Another person says Broker based and SAGAS
- Vertical scaling is mentioned in a meeting
- You are scared!



Tier 9

AI-Ops

- The AI-Ops people show up.
- ChatOps in Teams/Slack +1
- You are doing analysis and responses to customers using LLM tools like ChatGPT. +1
- **A Large language model writes automation code....**



SHODAN

Tier #FFFF FULL AI-CONTROL

- WHAT HAVE WE DONE?
- Powered by
ChatGPT™



Sentient Hyper-Optimized Data Access Network

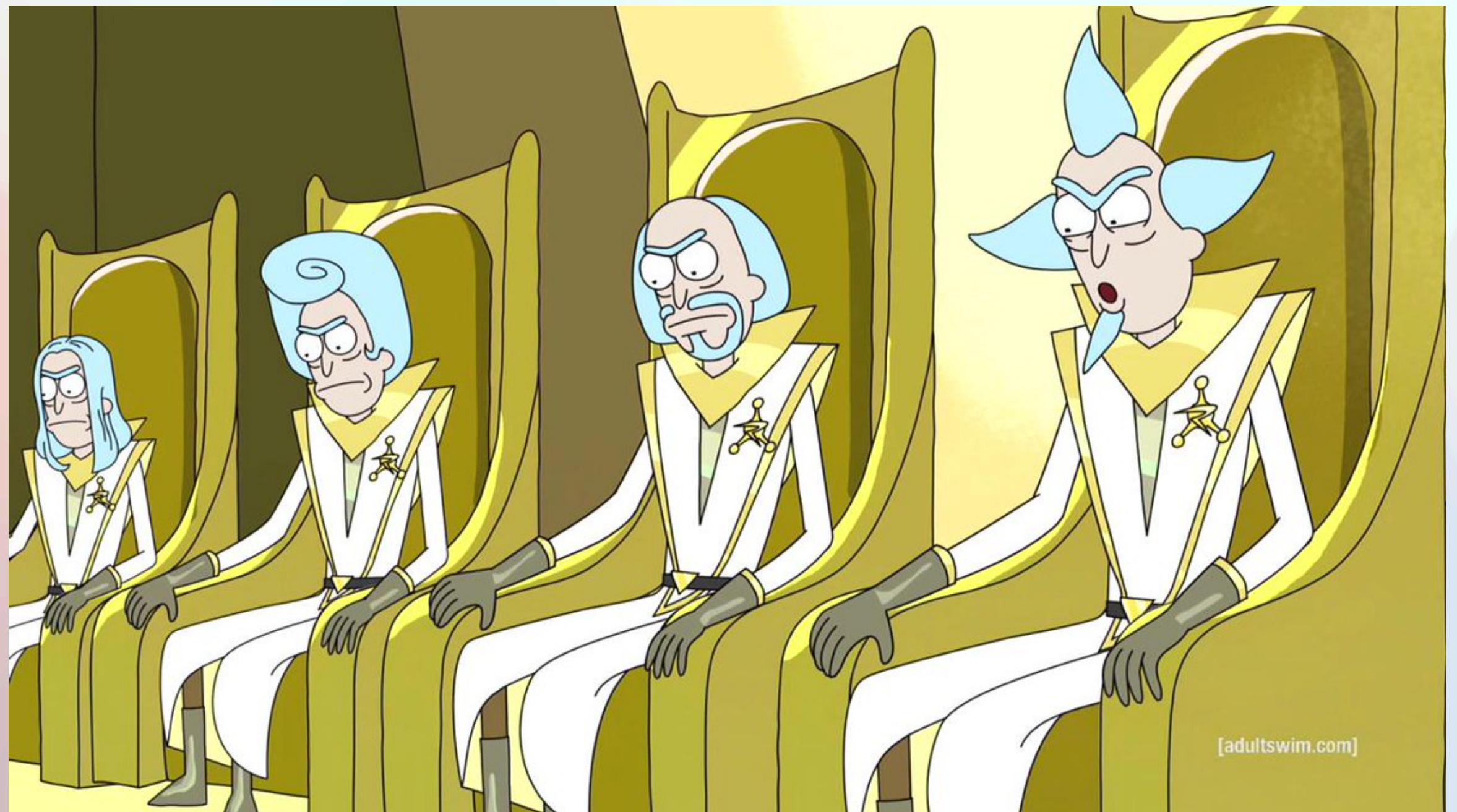
The background of the slide features a subtle, abstract design composed of numerous overlapping, semi-transparent colored layers. These layers create a soft, blurred effect with colors transitioning from light blue and green at the bottom to pink, orange, and yellow at the top. The overall aesthetic is clean and modern, providing a professional yet visually interesting backdrop.

Lessons Learned & Recommended resources!

Automation council

You don't know what you don't know!

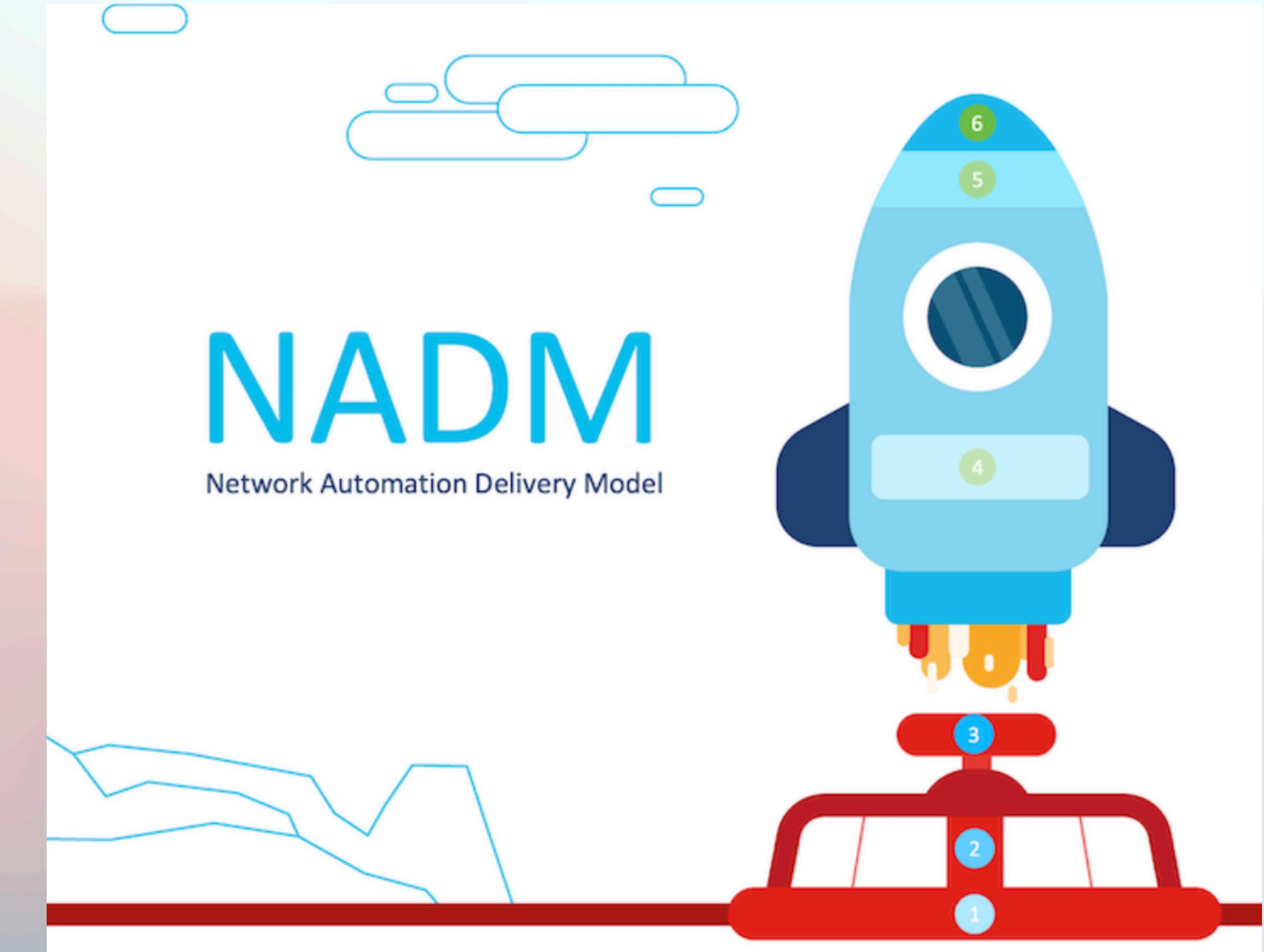
- Create an automation council!
- Multiple disciplines!
- Regular meetings and approval process!



NADM

Network Automation Delivery Model

- An excellent starting point for planning your automation journey!
- It's not a step-by-step guide.
- Mix it with other resources!



Define

Demonstrate

Deploy

Expand

Operationalize

Run

PHIND

The AI search engine for developers.

- It's very hit or miss like all LLM
- Worth checking out!
- Sites sources!!!



Can you make me a netbox script that adds a device?

Use Best Model (25 left)

Short Answer

Can you make me a netbox script that adds a device?

ANSWER | GPT-3.5 MODEL

To create a script that adds a device to NetBox, you can use the provided code snippet from gist.github.com. This script uses the `pynetbox` module to interact with the NetBox API and perform the necessary operations.

Here are the steps to create the script:

1. Install the `pynetbox` module by running the following command:

ChatGPT Netbox

life hack

- Don't tell it to make your plugin or script.
- Get a pseudocode language agreed on, then describe what you need using it.
- Way better results!
- It's not perfect, but it's better!



Create a pseudocode language for creating Netbox plugin



Netbox is an open-source web application designed to help manage and document computer networks. It supports creating plugins to extend its functionality. Although Netbox is developed using Python, the pseudocode below describes the general steps for creating a Netbox plugin, irrespective of the language.

sql Copy code

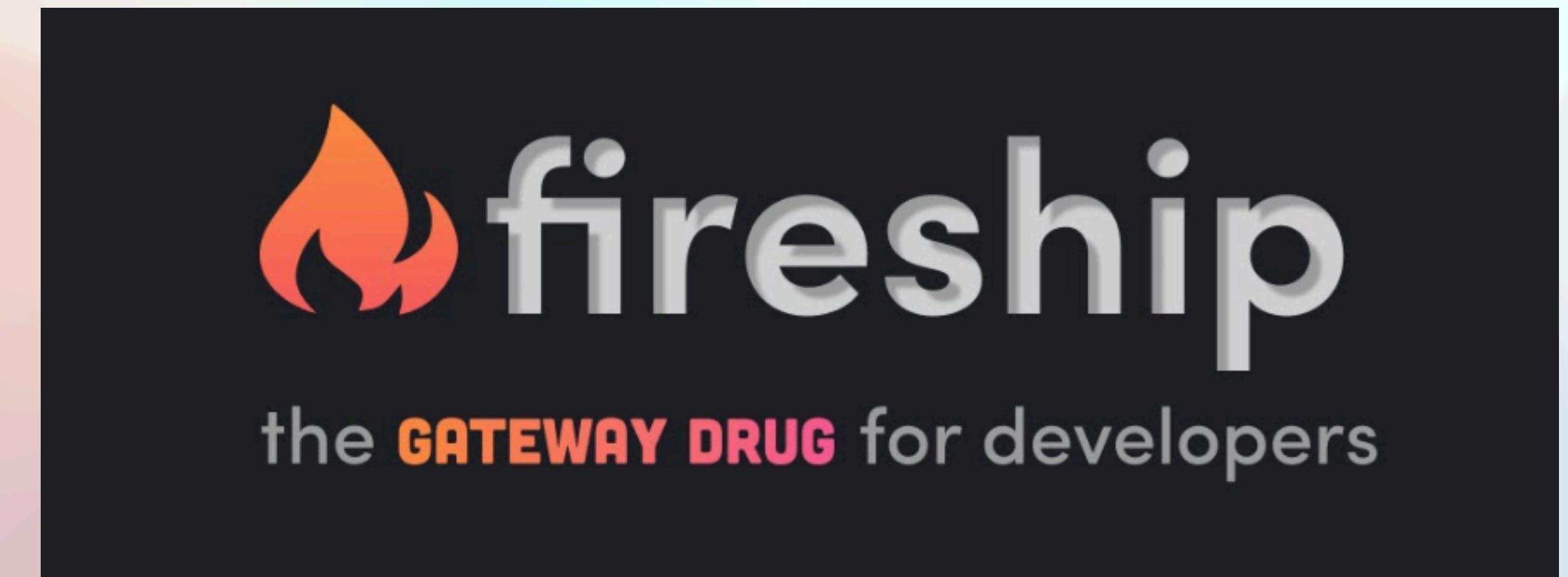
Create Netbox Plugin Pseudocode

1. Create Plugin Structure
 - a. Define Plugin name
 - b. Define Plugin version
 - c. Define Plugin description
 - d. Define required Netbox version
2. Define Plugin Configuration
 - a. Set Plugin settings (like custom fields, tables, etc.)
 - b. Set Plugin default settings (if any)
3. Create Plugin Code
 - a. Define the models (data structures) required by the plugin
 - i. Identify the fields and their types for each model
 - ii. Define any relationships between models (foreign key, many-to-many)
 - iii. Create appropriate methods for the models
 - b. Define the views (how the data is presented)
 - i. Identify the information to be shown in each view
 - ii. Define how users can interact with the data (create, update, delete)
 - c. Define the URLs (how the views are accessed)
 - i. Create URL patterns for each view
 - ii. Map each URL pattern to the appropriate view
 - d. Define the templates (the HTML structure of the views)

Fireship

Youtube

- Great AI/Dev framework news, not network automation focused at all!
- [https://www.youtube.com/
@Fireship](https://www.youtube.com/@Fireship)



Fundamentals of Software Architecture

by Mark Richards, Neal Ford

- Excellent overview of modern software architecture.
- Audiobook is available
- This is a great place to go if you don't know what microservices are!



**Fundamentals of
Software
Architecture**

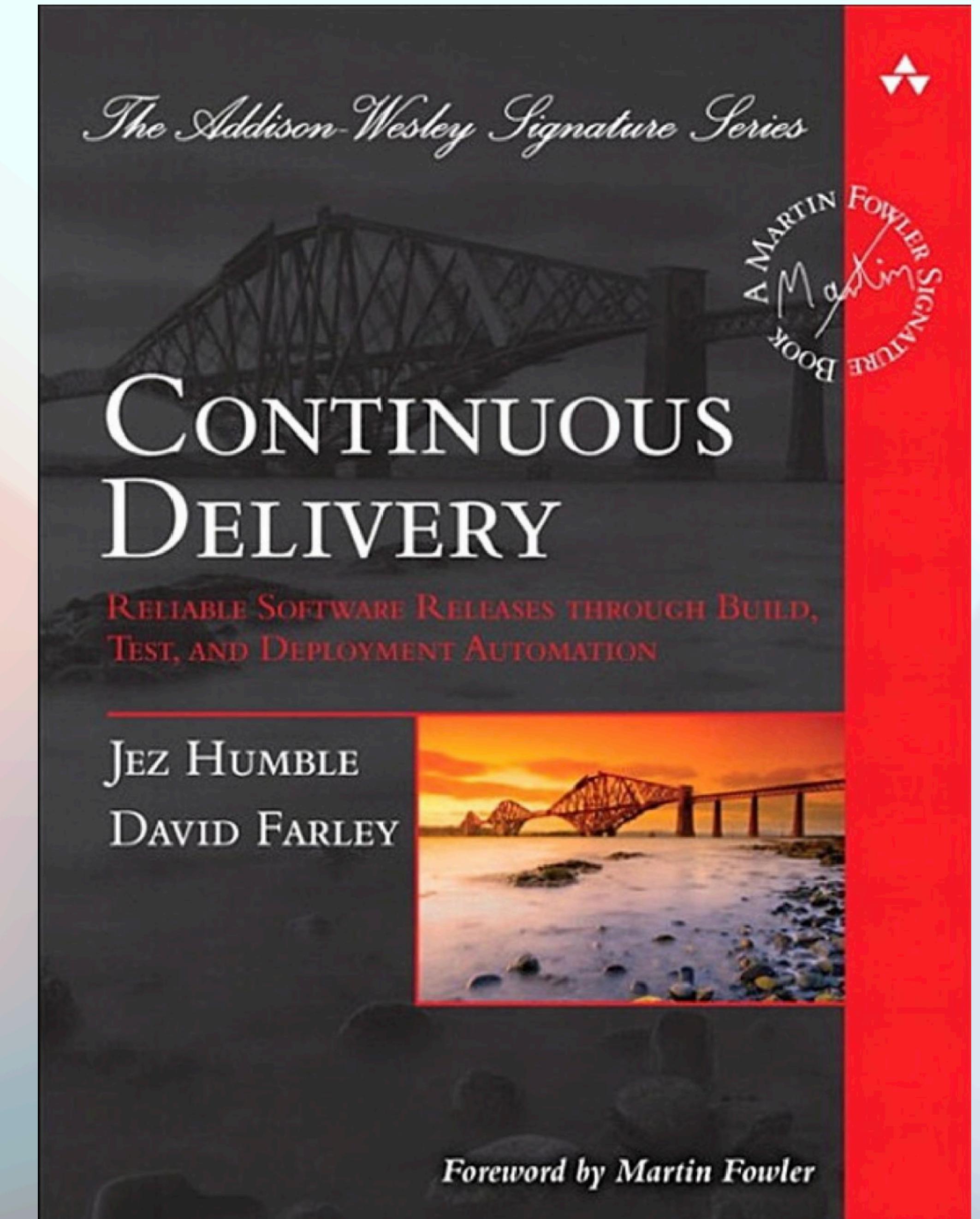
An Engineering Approach



Continuous Delivery

CI/CD meat and potatoes!

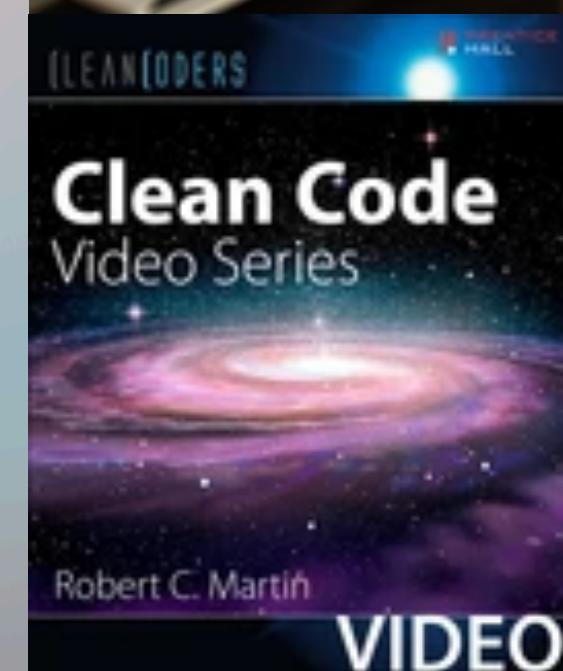
- My favorite book I read during my university study!
- The focus is on software development but applies to all aspects of service delivery.



Uncle Bob!

Robert C. Martin

- Excellent fundamentals of programming
- Clean code series
- Design Patterns series
- Yes, he is a Java guy; the ideas are essential, not the language.
- Go in with an open mind...
- Safari books is the best place to watch these videos!

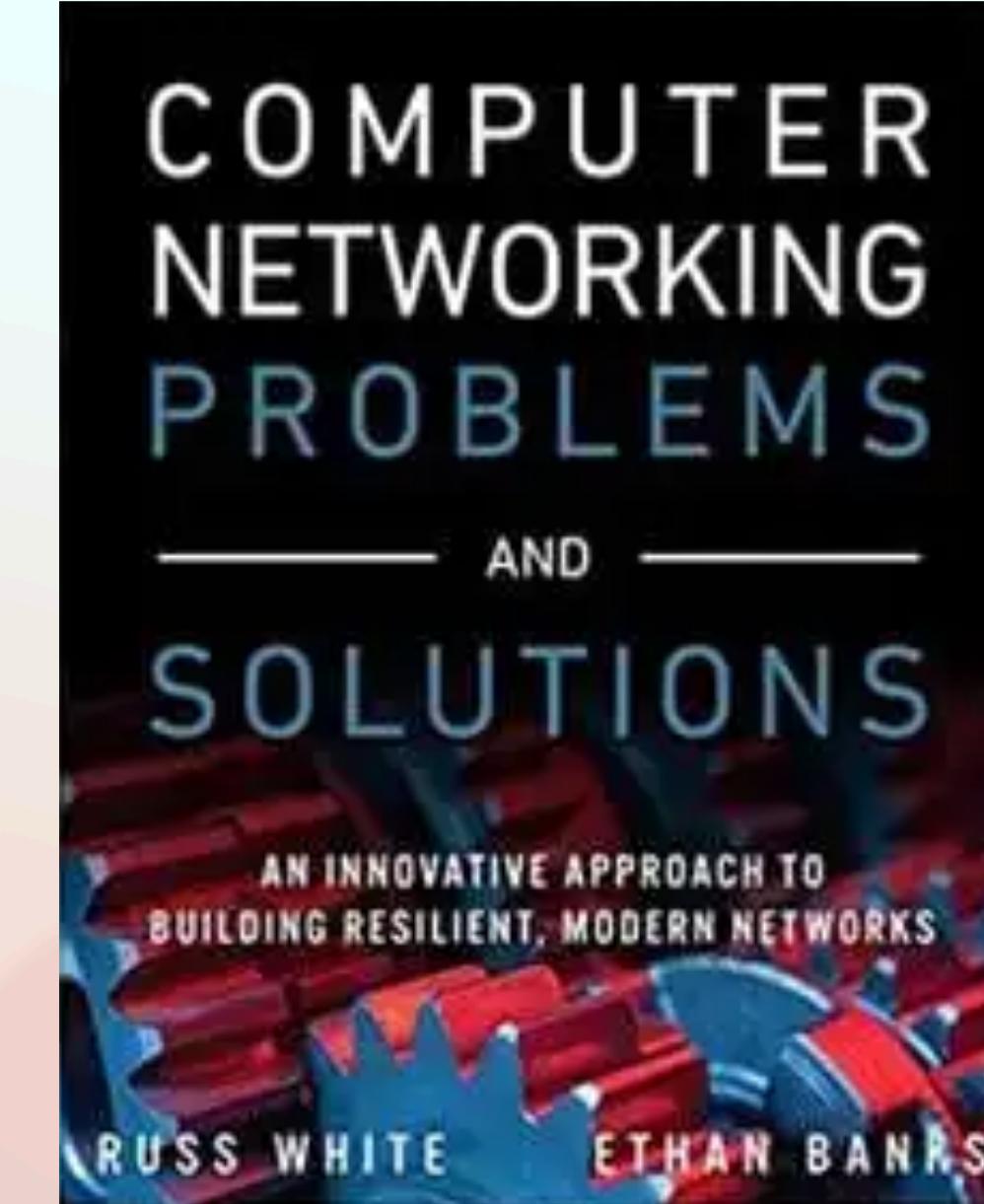


O'REILLY®
Safari

Russ White

The architects architect

- Podcasts
 - The hedge
 - History of networking
 - Can't fit all the stuff on a slide!



HISTORY OF
networking
THE PAST IS PRELUDE



Nick Russo

Great Network -> NetDev courses

- Very to the point, zero time wasted in his content!
- Everything from:
 - Git
 - Ansible
 - and beyond!

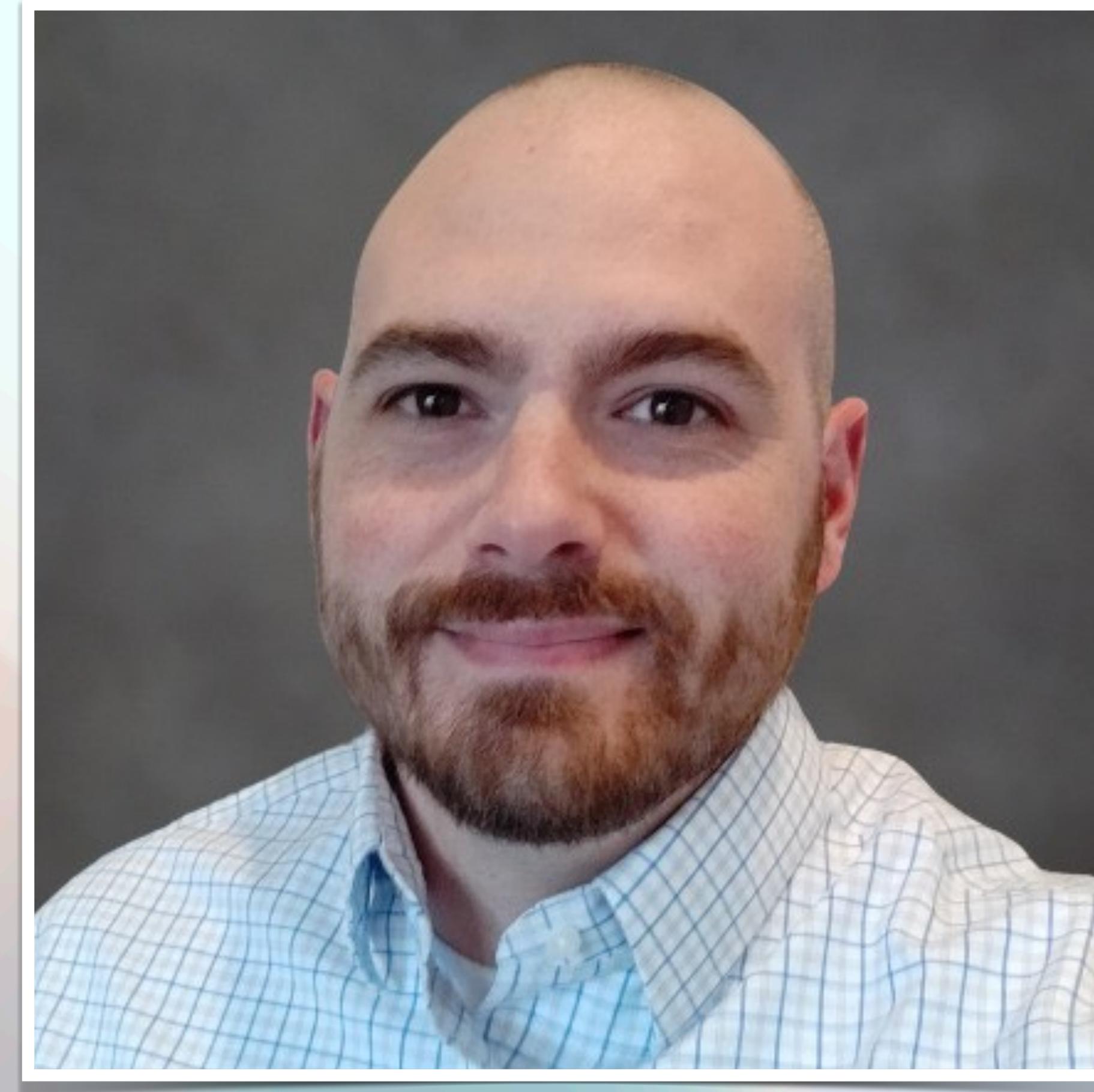


Last but certainly not least!

Jeremy Stretch

NetBox & PacketLife guy legend!

- Seven years grinding making Netbox what it is!
- packetlife.net since 2008 is one of the best CCIE resources!
 - Pcaps when I really needed them!
 - Blogs on exactly what I needed when I needed it!
- Absolute legend and the reason we are all here today!



netbox labs



Conclusion

Don't panic

Well maybe a little

- Things are evening out in the network automation space
- You don't need to invent the wheel
- There are off-the-shelf options for many things, but your business processes are still the main area of value!





LLM / “AI”

Means we should get more ambitious!

Thanks for listening!

Q&A

SLIDES!



Fun Ending

Switch rip and replace, example

Brownfield customer switch replacement

One way to do it.

- Gather information about the existing switch
 - Port counts
 - PoE budget etc.
- Go and configure line-by-line matching VLAN/Port assignments on the new switch
- Rack and carefully move all the connections from the old switch to the new switch

The background features a series of overlapping, rounded bands in shades of orange and yellow, creating a sense of depth and motion. The bands curve from the top left towards the bottom right.

Don't do this in production!

**Another way
to do this....**

Warning

This is a bad idea!

FEED CHATGPT!

ARP / Mac tables

Can you give me a list of OUI to VLAN mappings?

Here you go!

VLAN	OUI (First 6 characters)	Vendor
84	001871	Hewlett Packard Enterprise
	70b3d5	Dell Inc.
	001b4f	Avaya
85	70b3d5	Dell Inc.
	1c1d86	Cisco Systems, Inc.
	001b4f	Avaya
101	3cfdfc	Intel Corporate
230	2c3033	NETGEAR
	00e0fc	HPE
	001b4f	Avaya
243	001871	Hewlett Packard Enterprise
	00e0fc	HPE

Have it generate a template



```
router01#show arp dynamic
Protocol Address Age (min) Hardware Addr Type Interface
Internet 172.18.120.13 - a493.4c22.1111 ARPA GigabitEthernet0/1.3006
Internet 172.18.120.14 174 d46d.5025.2222 ARPA GigabitEthernet0/1.3006
Internet 172.18.120.62 2 a493.4c22.3333 ARPA GigabitEthernet0/0.84
Internet 172.18.120.222 0 a493.4c22.4444 ARPA GigabitEthernet0/0.85
Internet 172.18.124.21 96 3c08.f617.5555 ARPA GigabitEthernet0/0.101
```



curl://

```
1 curl --location --request PUT 'https://api.eu.mist.com/
  api/v1/orgs/nononono/networktemplates/
  96400f12-87ff-4975-a6ae-f80b84558abc' \
2 --header 'Accept: application/json' \
3 --header 'Content-Type: application/json' \
4 --header 'Authorization: Token nononono' \
5 --header 'Cookie: csrftoken.
  eu=nIM1Swy5BL9TGPjUXhg9qCDZxvn552rqzwDqfx6ypS7RzXq
  doQfY8k5jM0t26m' \
6 --data '{
  "port_usages": {
    "dynamic": {
      "mode": "dynamic",
      "rules": [
        {
          "src": "link_peermac",
          "usage": "84",
          "equals": "00:18:71",
          "expression": "[0:8]"
        },
        {
          "src": "link_peermac",
          "usage": "84",
          "equals": "70:B3:D5",
          "expression": "[0:8]"
        },
        {
          "src": "link_peermac",
          "usage": "84",
          "equals": "00:1B:4F",
          "expression": "[0:8]"
        }
      ]
    }
  }
}'
```



```
"dynamic": {
  "mode": "dynamic",
  "rules": [
    {
      "src": "link_peermac",
      "usage": "84",
      "equals": "00:18:71",
      "expression": "[0:8]"
    },
    {
      "src": "link_peermac",
      "usage": "84",
      "equals": "70:B3:D5",
      "expression": "[0:8]"
    },
    {
      "src": "link_peermac",
      "usage": "84",
      "equals": "00:1B:4F",
      "expression": "[0:8]"
    }
  ]
}
```



Can you give me a list of OUI to VLAN mappings?



Here you go!

VLAN	OUI (First 6 characters)	Vendor
84	001871	Hewlett Packard Enterprise
	70b3d5	Dell Inc.
	001b4f	Avaya
85	70b3d5	Dell Inc.
	1c1d86	Cisco Systems, Inc.
	001b4f	Avaya
101	3cfdf	Intel Corporate
230	2c3033	NETGEAR
	00e0fc	HPE
	001b4f	Avaya
243	001871	Hewlett Packard Enterprise
	00e0fc	HPE

Home Workspaces API Network Explore Search Postman Invite Settings Upgrade

MIST_TEST_WORKSPACE_V01 New Import Overview Mist Cloud Mist Cloud GET getOrgNetw GET getOrgDe PUT updateOr advania poc + No Environment

Collections APIs Environments Mock Servers Monitors Flows History

+ / O.. Netw... / updateO... Save ...

PUT https://{{host}}/api/v1/orgs/:org_id/networkitem Send

Params Auth Headers (14) Body Pre-req. Tests Settings

raw JSON Beautify

1 {
2 "port_usages": {
3 "dynamic": {
4 "mode": "dynamic",
5 "rules": [
6 {
7 "src": "link_peermac",
8 "usage": "guest",
9 >equals": "00:18:71",
10 "expression": "[0:8]"
11 },
12 {
13 "src": "lldp_system_description",
14 "usage": "uplink",
15 >equals": "Juniper",
16 "expression": "[0:7]"
17 },
18 {
19 "src": "lldp_system_description",
20 "usage": "ap",
21 >equals": "Mist",
22 "expression": "[0:4]"
23 }
24 }
25 }
26 }
27 }

cURL

```
curl --location --request PUT 'https://api.eu.mist.com/api/v1/orgs/nononono/networktemplates/'  
--header 'Accept: application/json'  
--header 'Content-Type: application/json'  
--header 'Authorization: Token nononono'  
--header 'Cookie: csrftoken.'
```

--data '{
 "port_usages": {
 "dynamic": {
 "mode": "dynamic",
 "rules": [
 {
 "src": "link_peermac",
 "usage": "guest",
 >equals": "00:18:71",
 "expression": "[0:8]"
 },
 {
 "src": "lldp_system_description",
 "usage": "uplink",
 >equals": "Juniper",
 "expression": "[0:7]"
 },
 {
 "src": "lldp_system_description",
 "usage": "ap",
 >equals": "Mist",
 "expression": "[0:4]"
 }
 }
 }
 }
}

can you make mappings based on this table and make rules based on each mac OUI, i've already added the 00:18:71

```
"dynamic": {  
    "mode": "dynamic",  
    "rules": [  
        {  
            "src": "link_peermac",  
            "usage": "84",  
            "equals": "00:18:71",  
            "expression": "[0:8]"  
        },  
    ]  
}
```



Certainly! Here are the rules based on the table provided, excluding the rule for "00:18:71" like dislike
which you have already added:

```
css  
  
"dynamic": {  
    "mode": "dynamic",  
    "rules": [  
        {  
            "src": "link_peermac",  
            "usage": "84",  
            "equals": "00:18:71",  
            "expression": "[0:8]"  
        },  
        {  
            "src": "link_peermac",  
            "usage": "84",  
            "equals": "00:18:71",  
            "expression": "[0:8]"  
        }  
    ]  
}
```



Here is the updated CURL call with the new rules:

bash

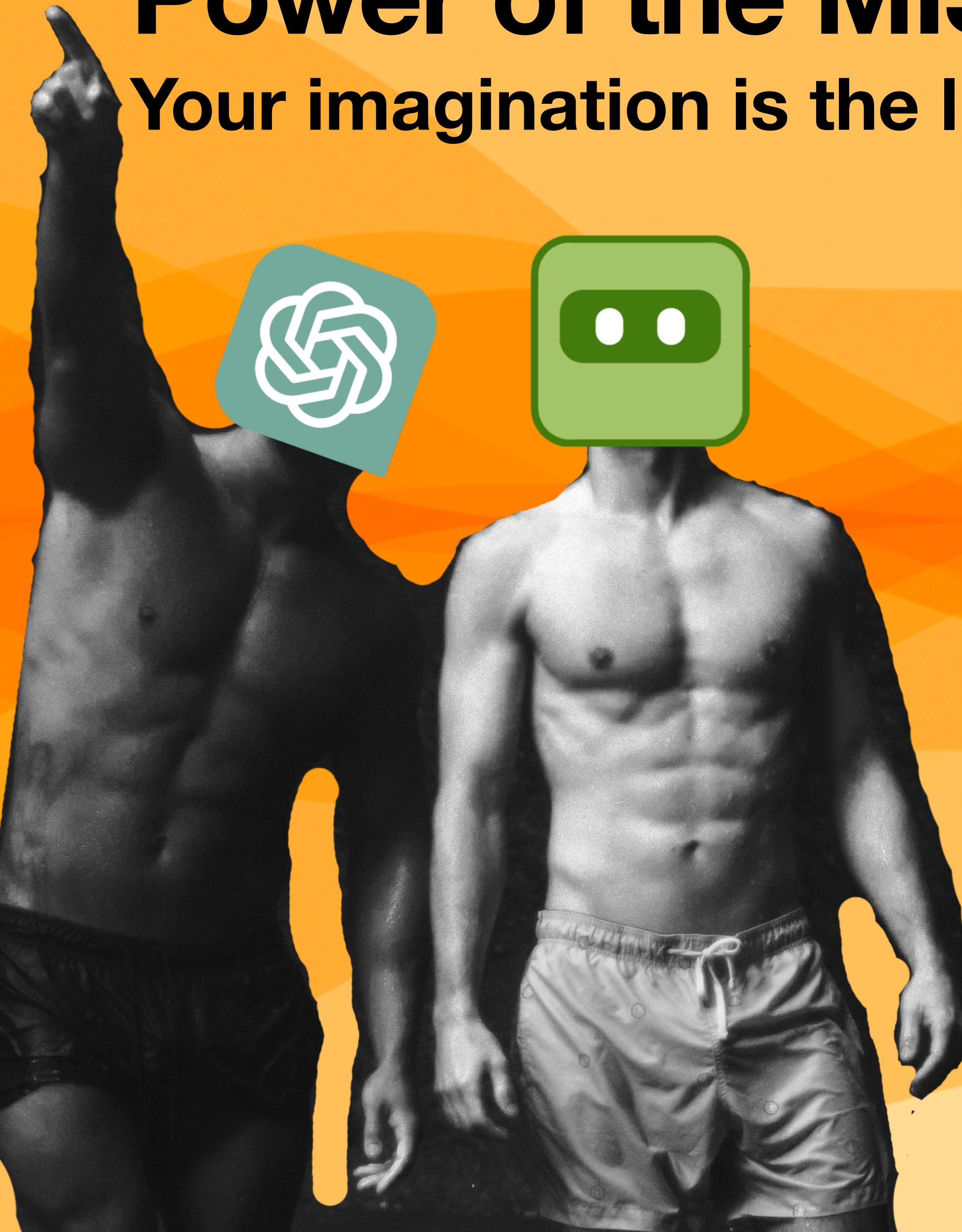
Copy code

```
curl --location --request PUT 'https://api.eu.mist.com/api/v1/orgs/nononono/ports/1000'
--header 'Accept: application/json' \
--header 'Content-Type: application/json' \
--header 'Authorization: Token nononono' \
--header 'Cookie: csrftoken.eu=nononono'
--data '{
  "port_usages": {
    "dynamic": {
      "mode": "dynamic",
      "rules": [
        {
          "src": "link_peermac",
          "usage": "84",
          "equals": "00:18:71",
          "expression": "[0:8]"
        },
        {
          "src": "link_peermac",
          "usage": "84",
          "equals": "70:B3:D5",
          "expression": "[0:8]"
        },
        {
          "src": "link_peermac",
          "usage": "84",
          "equals": "00:1B:4F",
          "expression": "[0:8]"
        }
      ]
    }
  }
}'
```

Power of the MIST API

Your imagination is the limit.

- We took a manual task
- Used an LLM to parse the data
- Invoked the power of the Mist API to do exactly what we wanted.
- Dynamic port configuration is doing the heavy lifting here.
- DON'T TRUST THE LLM LIKE YOU TRUST MARVIS!



Thanks for listening!

Q&A

SLIDES!

