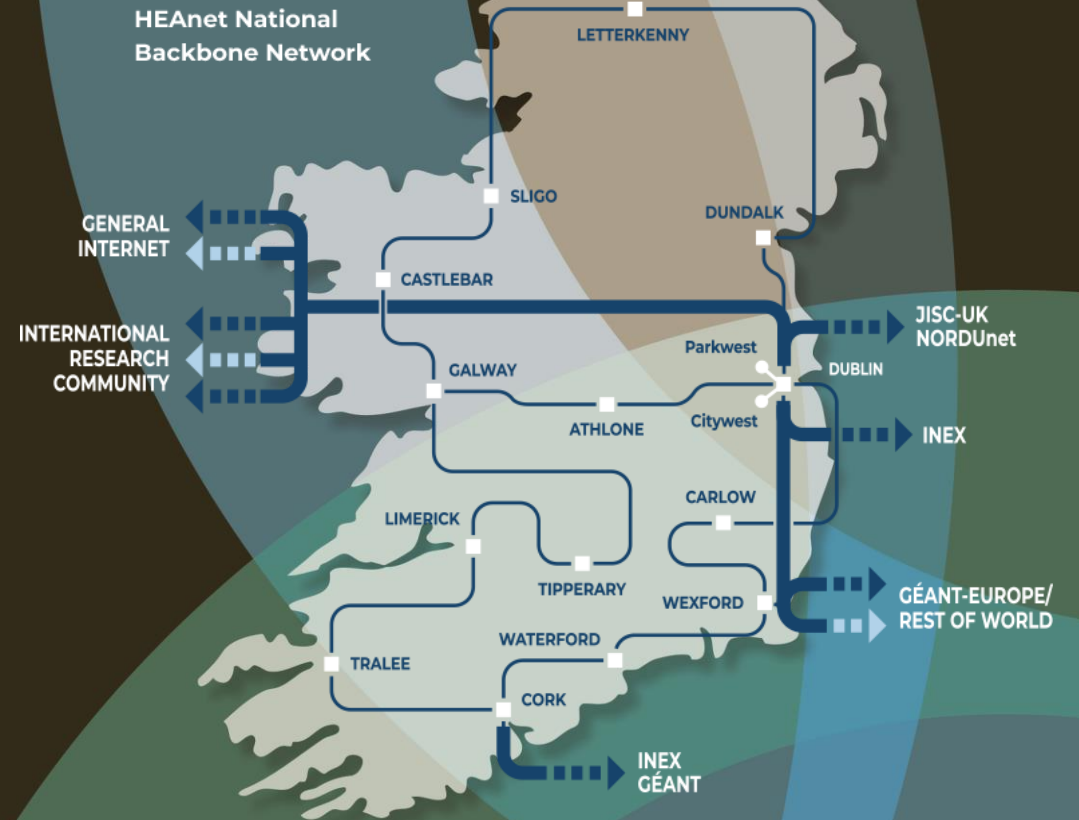




# HEAnet Automation Tools (HAT) Our future, orchestrated!

RESIN Talk April 2024

Andy Byrne & Brian McArdle



# Agenda

## Automation in HEAnet

- Why Automate
- Levels of Automation
- Teams Required
- Choices available
- Tools we use
- Timelines
- Starting an Automation Project

## Service Provisioning Project

Q & A

# Why Automate ?

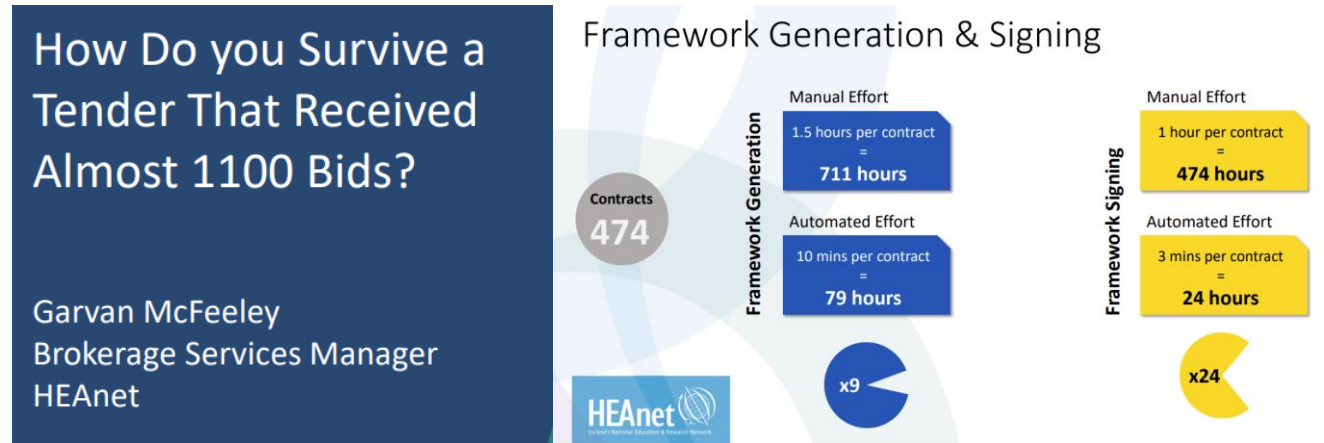
- Improvement on efficiency
- Reduction of human error
- Improvement of network availability
- Reduction of time wasted on repetitive tasks
- Reduction of operating expenses

# Different levels of automation

1. Simple Standalone scripts
  - One off tasks. Simple input/output
2. API-based automation scripting:
  - Replacing manual command-line instructions.
3. High-level programming languages to invoke APIs:
  - Python, Java, etc.
4. Machine learning and Big Data:
  - Allowing autonomous decision-making and configuration

# HEAnet & Automation

- Macros
- Advanced Excel/MS Tools
- <https://www.heanet.ie/wp-content/uploads/2022/02/Thursday-A-3.30-Garvan-McFeeley.pdf>



- Stand Alone Scripts
- In-house Middleware
- Commercial Off Shelf Offerings
- Fully developed advanced interdependent systems



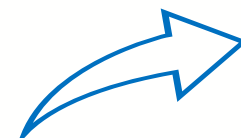
Schools Network



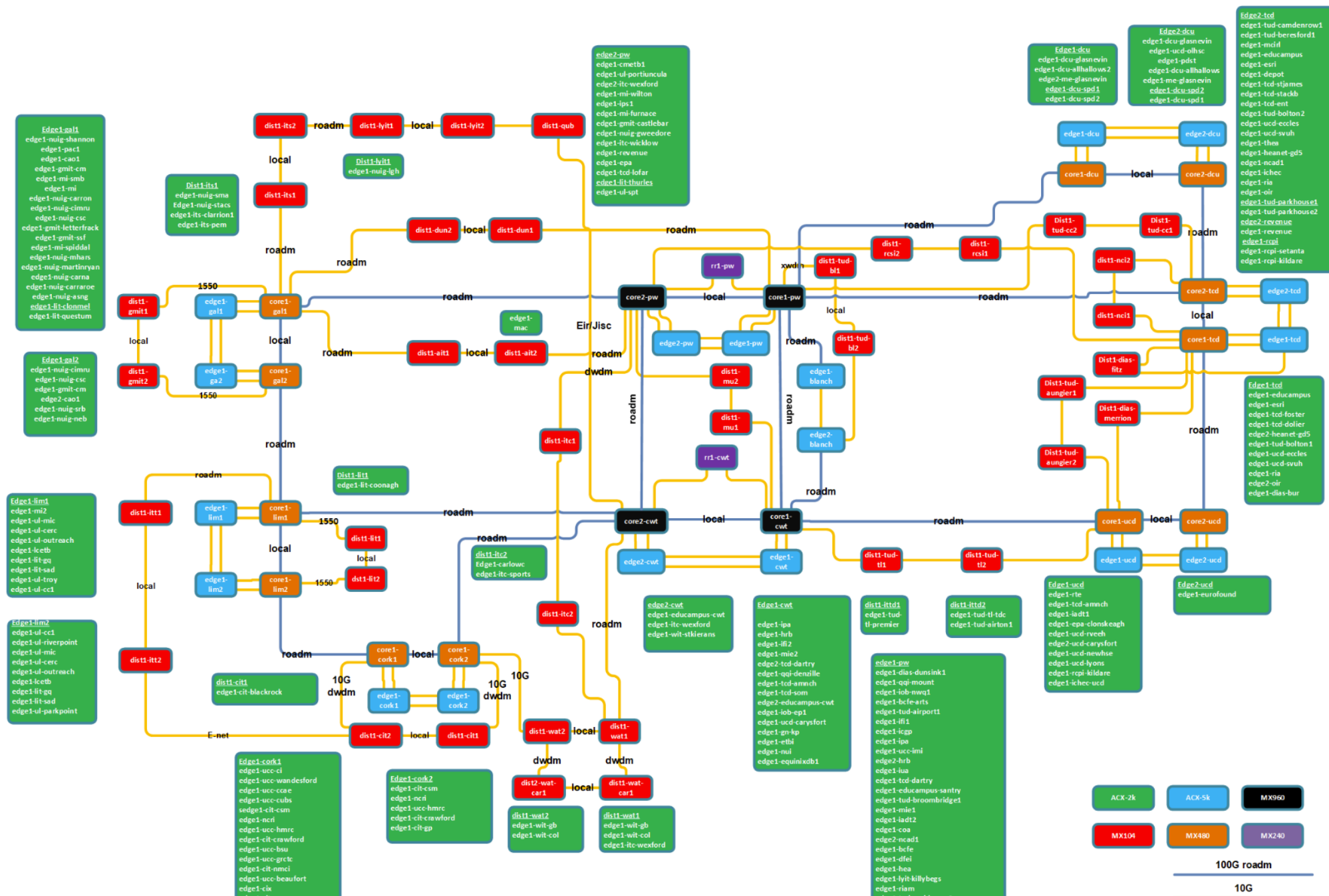
Shiba



Dynamics / Power Automate / Adobe Sign



Services Provisioning



Edge1-gal1  
 edge1-nuig-shannon  
 edge1-pact  
 edge1-ca01  
 edge1-gmit-cm  
 edge1-mi  
 edge1-nuig-carron  
 edge1-nuig-clmu  
 edge1-nuig-csc  
 edge1-gmit-letterfrack  
 edge1-gmit-ssf  
 edge1-mi-spiddal  
 edge1-nuig-mhars  
 edge1-nuig-martinyan  
 edge1-nuig-carna  
 edge1-nuig-carraroe  
 edge1-nuig-asng  
 edge1-it-clomml  
 edge1-it-questum

Edge1-gal2  
 edge1-nuig-clmu  
 edge1-nuig-csc  
 edge1-gmit-cm  
 edge2-ca01  
 edge1-nuig-srb  
 edge1-nuig-neb

Edge1-lim1  
 edge1-mi2  
 edge1-ul-mic  
 edge1-ul-cerc  
 edge1-ul-outreach  
 edge1-lcceb  
 edge1-lit-gg  
 edge1-lit-sad  
 edge1-ul-troy  
 edge1-ul-cci

Edge1-lim2  
 edge1-ul-cci  
 edge1-ul-riverpoint  
 edge1-ul-mic  
 edge1-ul-outreach  
 edge1-lcceb  
 edge1-lit-gg  
 edge1-lit-sad  
 edge1-ul-parkpoint

Edge1-cork1  
 edge1-ucc-cl  
 edge1-ucc-wandesford  
 edge1-ucc-ccae  
 edge1-ucc-cbs  
 edge1-clt-csm  
 edge1-ncr  
 edge1-ucc-hmrc  
 edge1-clt-crawford  
 edge1-ucc-bou  
 edge1-ucc-avtc  
 edge1-clt-nmci  
 edge1-ucc-beaufort  
 edge1-clk  
 edge1-clt-sp

Edge1-cork2  
 edge1-clt-csm  
 edge1-ncr  
 edge1-ucc-hmrc  
 edge1-clt-crawford  
 edge1-clt-gp

dist1-wat2  
 edge1-wit-gb  
 edge1-wit-col

dist1-wat1  
 edge1-wit-gb  
 edge1-wit-col  
 edge1-rtc-wexford

edge2-pw  
 edge1-cmetb1  
 edge1-ul-portiuncula  
 edge2-rtc-wexford  
 edge1-mi-wilton  
 edge1-lips1  
 edge1-mi-furnace  
 edge1-gmit-castlebar  
 edge1-nuig-gvevordc  
 edge1-rtc-wexford  
 edge1-revenue  
 edge1-epa  
 edge1-td-lofar  
 edge1-lit-thurles  
 edge1-ul-spt

Edge1-dcu  
 edge1-dcu-glasnevin  
 edge1-dcu-allhallows2  
 edge2-me-glasnevin  
 edge1-dcu-spd1  
 edge1-dcu-spd2

Edge2-dcu  
 edge1-tud-olivsc  
 edge1-pd01  
 edge1-me-glasnevin  
 edge1-dcu-spd2  
 edge1-dcu-spd1

Edge2-tdc  
 edge1-tud-camdenrow1  
 edge1-tud-beresford1  
 edge1-mcid  
 edge1-educampus  
 edge1-esri  
 edge1-depot  
 edge1-td-stjames  
 edge1-td-stackb  
 edge1-td-sent  
 edge1-tud-bolton2  
 edge1-ucc-eccles  
 edge1-ucc-svuh  
 edge1-thea  
 edge1-heatet-gd5  
 edge1-ncad1  
 edge1-ichec  
 edge1-ria  
 edge1-oir  
 edge1-tud-parkhouse1  
 edge1-tud-parkhouse2  
 edge2-revnuic  
 edge1-revenue  
 edge1-zanl  
 edge1-rp1-setanta  
 edge1-rp1-kildare

Edge1-tdc  
 edge1-educampus  
 edge1-esri  
 edge1-td-foster  
 edge1-td-dobler  
 edge2-heatet-gd5  
 edge1-tud-bolton1  
 edge1-ucc-eccles  
 edge1-ucc-svuh  
 edge1-ria  
 edge2-oir  
 edge1-dias-bur

ACX-2k ACX-5k MX960

MX104 MX480 MX240

100G roadm

10G

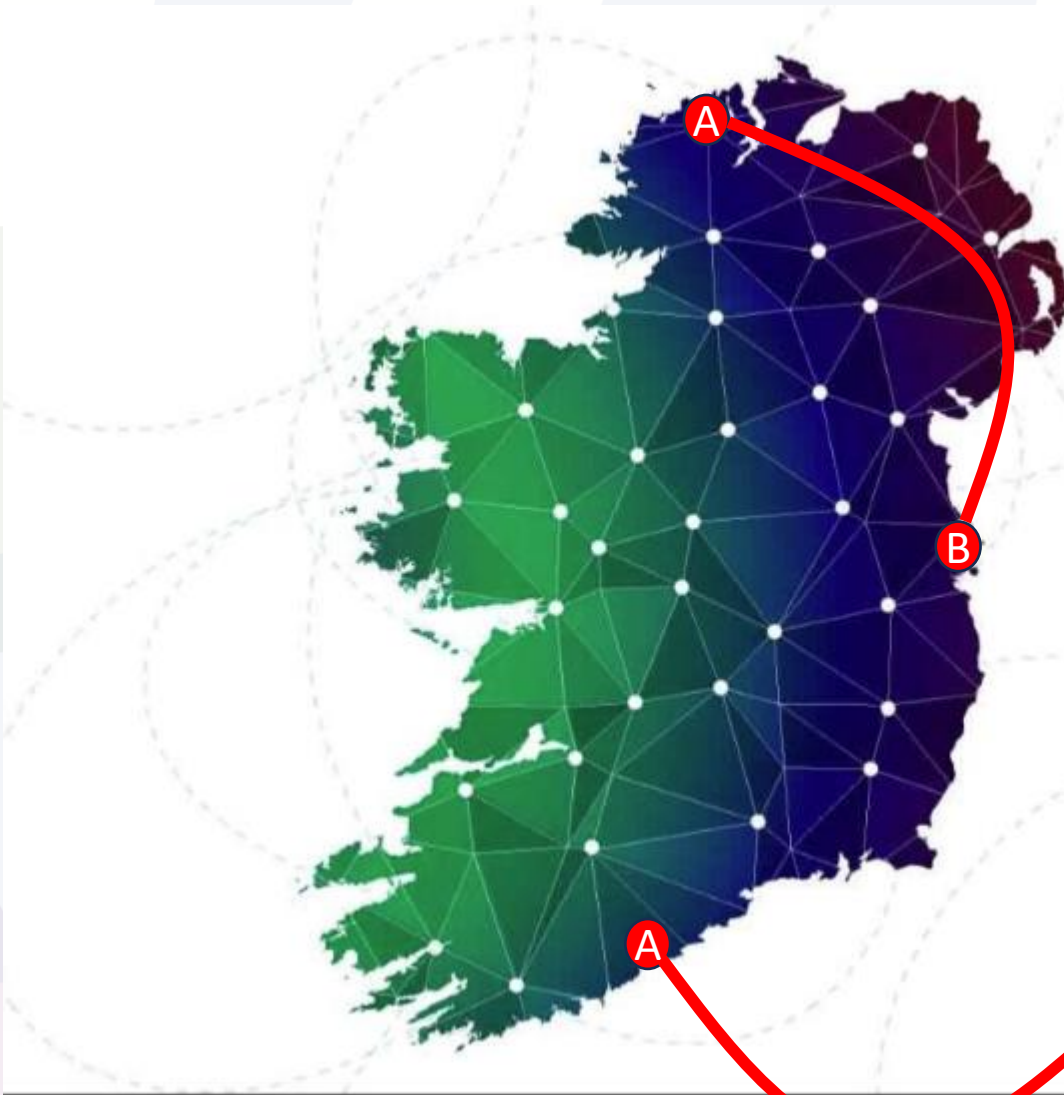
# Configure Routers

## - Service Provisioning Project

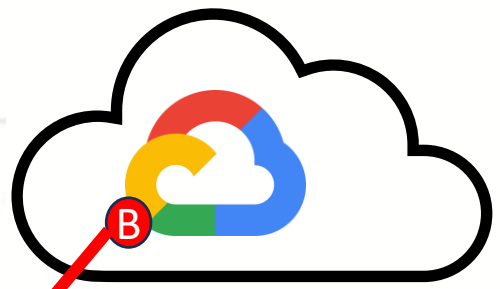
The screenshot shows the Junos Services Director interface. At the top, there are navigation tabs for Views (Service View), Task Categories (Build, Deploy, Monitor, Fault, Report), and Connectivity Services Director Banner. The main area is divided into a Tasks pane on the left and a Main Window on the right. The Tasks pane lists various tasks like 'Manage Services', 'Manage Service Definitions', and 'Manage Customers'. The Main Window displays three pie charts: 'Services by Customer', 'Services per Device', and 'Services by Type'. Below the charts is a table of service instances with columns for Name, Service Type, Customer, State, FA Status, Fault Status, SLA Status, PM Status, Definition, Activation Date, and Last Modified. An 'Alarms' section at the bottom shows a summary of alarm counts: 1 Critical, 1 Major, 0 Minor, and 0 Info.

Name	Service Type	Customer	State	FA Status	Fault Status	SLA Status	PM Status	Definition	Activation Date	Last Modified
VPLS_OnQ_AL...	VPLS	abcd	Deployed-Active	Down	Up	None	None	ELAN-BGP-Qn...	June 11, 2015, ...	June 11, 2015, ...
P2P_LDP_100	P2P	abcd	Deployed-Active	Pending	Up	None	None	ELLine-Dot1a-Sl...	June 11, 2015, ...	June 12, 2015, ...
prakasidestp2...	P2P	abcd	Deployed-Active	Pending	Up	None	None	ELLine-BGP-Dot...	June 8, 2015, ...	June 8, 2015, ...
Test_Eline_BG...	P2P	ATT	Deployed-Active	Down	Down	None	None	ELLine-BGP-Dot...	June 11, 2015, ...	June 11, 2015, ...
P2P_BGP_res1	P2P	abcd	Deployed-Active	Down	Up	None	None	P2P_reslency...	June 11, 2015, ...	June 11, 2015, ...
dsadsa	P2P	ATT	Deployed-Active	Pending	None	None	None	BOPSD_Templ	June 12, 2015, ...	June 12, 2015, ...





```
protocols {  
  l2circuit {  
    neighbor {{ l2circuit.p2p_endpoint }} {  
      interface {{ l2circuit.ce_ifl }} {  
        virtual-circuit-id {{ l2circuit.vc_id }};  
        mtu 9192;  
        pseudowire-status-tlv;  
      }  
    }  
  }  
}
```



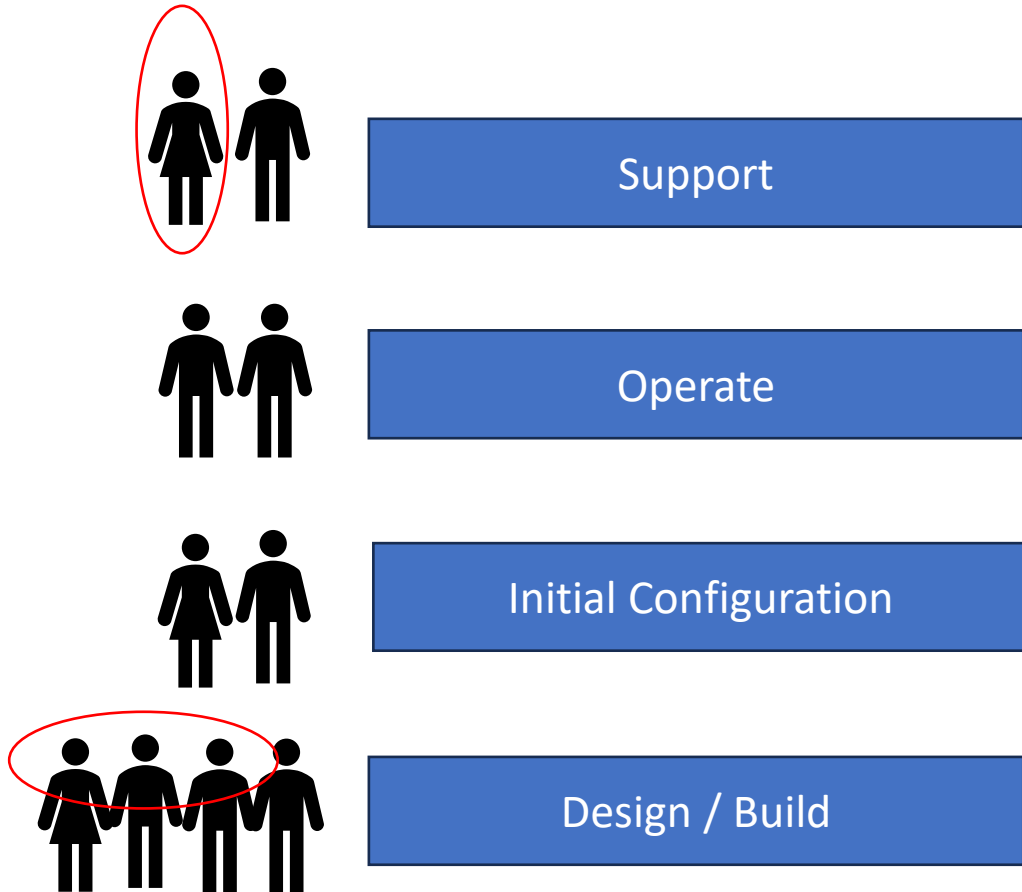
L2VPN



# Anatomy of a Service

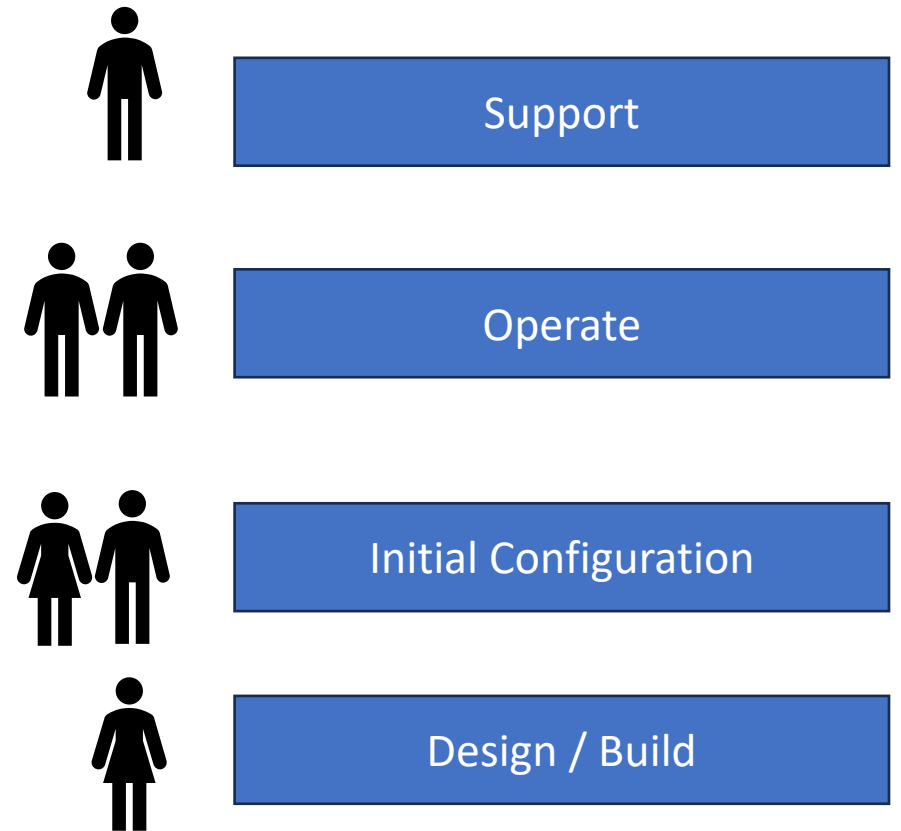
Plan A

Build



Plan B

Buy



# Pros & Cons Build your own model

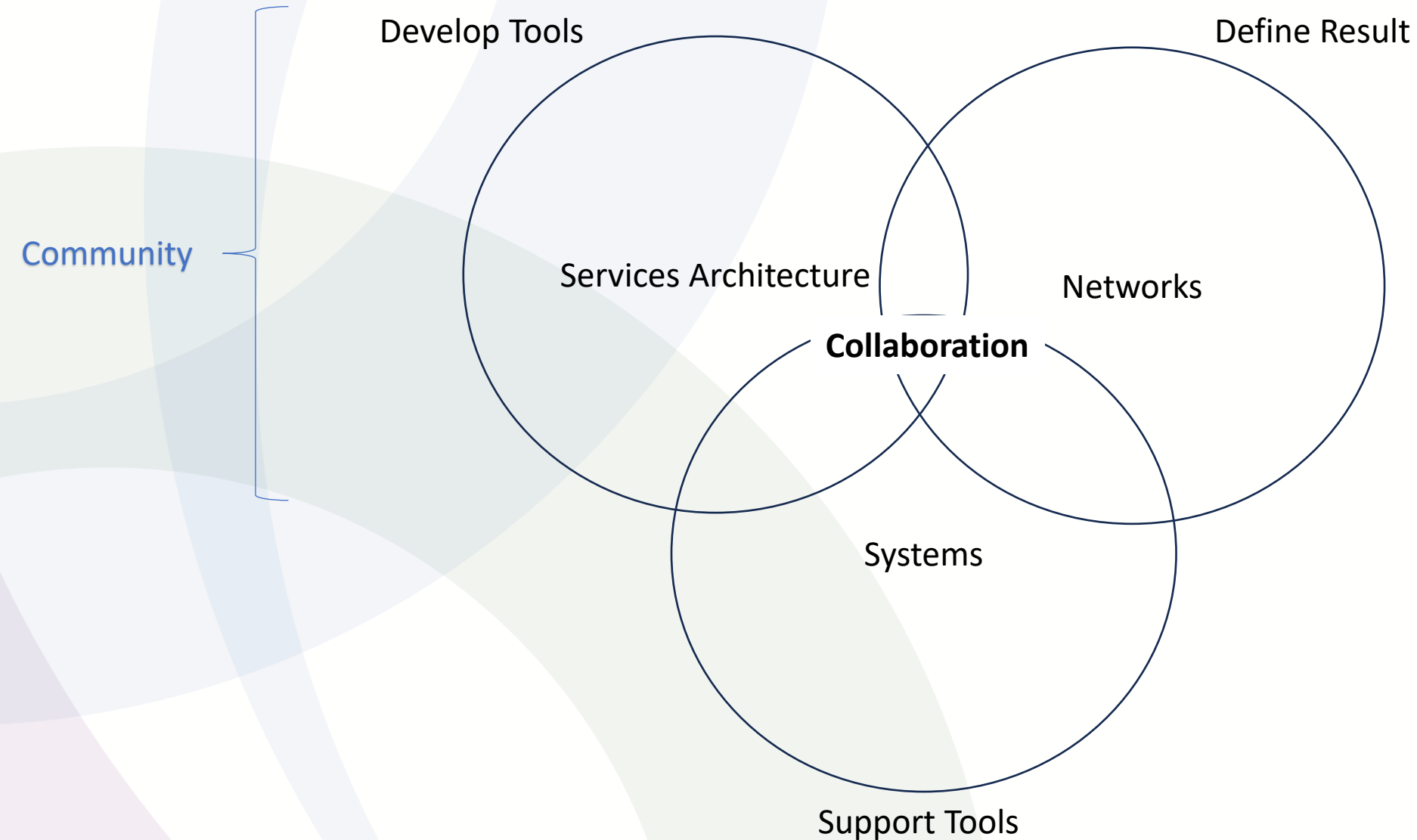
- Pros

- Manage Lifecycle
- Manage Cost
- Multivendor
- Extensible

- Cons

- Takes longer
- Learning curve
- More Responsibility

# HEAnet Technical Teams - Automation



# Single Source of Truth

- One information repository
- Model all data
- Open Inputs/Outputs
- Audited Changes



# Netbox – Single Source of Truth

Name	Status	Tenant	Site	Location	Rack	Role	Manufacturer	Type	IP Address
annaw-bulk-1	Active	HEAnet	BT Citywest	—	—	Edge Router	Juniper	ACX5048-AC	—
annaw-bulk-2	Active	HEAnet	BT Citywest	—	—	Edge Router	Juniper	ACX5048-AC	—
annaw101	Active	HEAnet	North Dock Two	—	—	Core Router	Juniper	MX240	87.44.48.199/32
bma-local-please-dont-delete-me	Active	HEAnet	North Dock Two	—	—	Edge Router	Juniper	ACX2200-AC	87.44.48.3/32
core1-cork1	Active	HEAnet	UCC PoP1	—	—	Core Router	Juniper	MX480	87.44.48.40/32
core1-cvt	Active	HEAnet	BT Citywest	—	—	Core Router	Juniper	MX960 With Cable Management	87.44.48.1/32
core1-pw	Active	HEAnet	Parkwest   Intersion	—	—	Core Router	Juniper	MX960 With Cable Management	87.44.48.2/32
core2-cvt	Active	HEAnet	BT Citywest	—	—	Core Router	Juniper	MX960 With Cable Management	87.44.48.24/32
core2-pw	Active	HEAnet	Parkwest   Intersion	—	—	Core Router	Juniper	MX960 With Cable Management	87.44.48.25/32
dist1-tud-b42	Active	HEAnet	TUD Blanchardstown PoP2	—	—	Dial Router	Juniper	MX104-PREMIUM	87.44.48.8/32
edge1-cix	Active	HEAnet	Cork Internet Exchange	—	CIX1BK09	Edge Router	Juniper	ACX2200-AC	87.44.48.4/32
edge1-clisix	Active	HEAnet	BT Citywest	—	—	Edge Router	Juniper	ACX2200-AC	87.44.48.59/32
edge1-conference-killarney	Active	HEAnet	Conference Hotel - Great Southern Killarney	—	—	Edge Router	Juniper	ACX2200-AC	87.44.49.15/32
edge1-cvt	Active	HEAnet	BT Citywest	—	—	Edge Router	Juniper	ACX5048-AC	87.44.48.11/32
edge1-dcu	Active	HEAnet	DCU PoP1	—	—	Edge Router	Juniper	ACX5048-AC	87.44.48.14/32
edge1-dcu-morton1	Active	HEAnet	DCU Morton Stadium	—	—	Edge Router	Juniper	ACX2200-AC	87.44.49.19/32
edge1-dcu-spd1	Active	HEAnet	DCU St. Patrick's Drumcondra PoP1	—	—	Edge Router	Juniper	ACX5048-AC	87.44.48.16/32
edge1-greth-boyle	Active	Galway Roscommon ETB	GRETB Boyle FET	—	—	Edge Router	Juniper	ACX2200-AC	—
edge1-heanet-nd2	Active	HEAnet	North Dock Two	—	—	Edge Router	Juniper	ACX7024-AC	87.44.48.234/32
edge1-ipa	Active	HEAnet	Institute of Public Administration	—	—	Edge Router	Juniper	ACX2200-AC	87.44.48.226/32
edge1-mic-courtbrack1	Active	HEAnet	MIC Courtbrack Student Accommodation	—	—	Edge Router	Juniper	ACX2200-AC	87.44.49.29/32
edge1-servprov-testlab	Active	HEAnet	North Dock Two	—	—	Edge Router	Juniper	ACX2200-AC	87.44.75.254/32
edge2-blanch	Active	HEAnet	Blanchardstown   DRT	—	—	Edge Router	Juniper	ACX5048-AC	87.44.48.39/32
edge2-clisix	Active	HEAnet	BT Citywest	—	—	Edge Router	Juniper	ACX2200-AC	87.44.48.241/32

## Hierarchy of Data

- Devices
- Platforms
- Configuration
- L2VPNs

## Added bonus!

- IP Address Management
- Data Centre Inventory

# Suite of Network Automation Tools

- Ansible Playbooks
  - Command Line Interface
  - No Inventory
  - Steps involved are not clear
- Napalm
- Gitlab

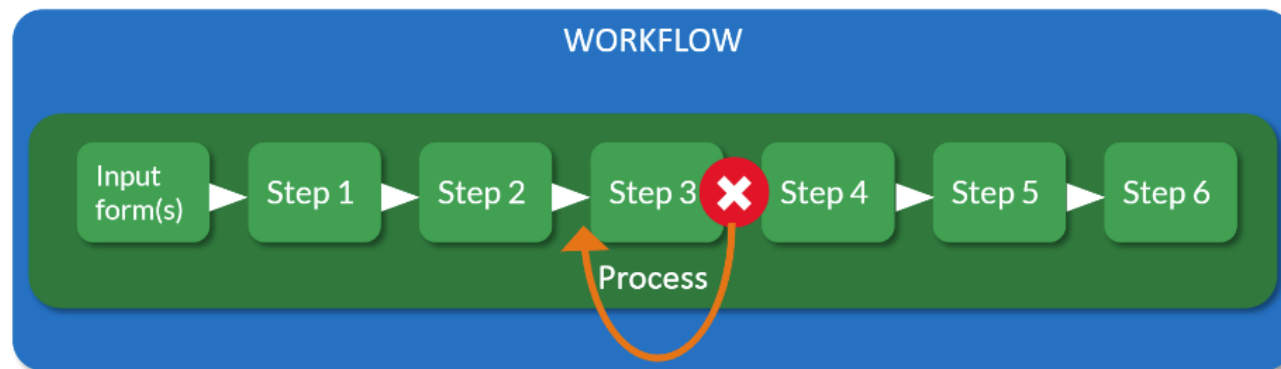
# What's Missing ?

Single Source Of Truth

Suite of Automation Tools :  
Ansible  
Gitlab  
Napalm

??

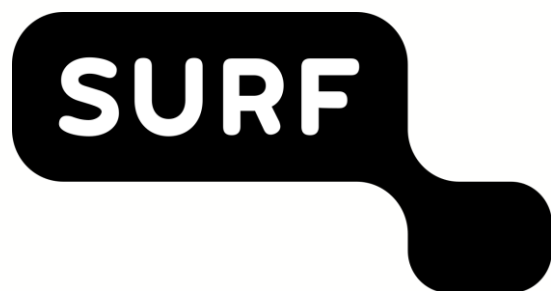
Workflow Engine



- Each Step writes the state to the database and is used as input for the next step
  - Each (atomic) Step can be retried, making the workflow robust



# Workflow Orchestrator



**ESnet**

ENERGY SCIENCES NETWORK



**Node Settings**

**Node Type \***

Juniper ACX2200-AC

**Node Role \***

Edge Router

**Site \***

Select Placeholder

ATU Castlebar

ATU Cluain Mhuire

## I2vpn

### Ports \*

Select Placeholder

port 1G edge2-servprov-testlab ge-0/0/0 I2vpn-to-edge1

port 1G edge1-servprov-testlab ge-0/0/0 I2vpn-to-edge2

port 1G edge1-servprov-testlab ge-0/0/0 (UN) edge1 1G port - p2p services testing with callbacks

port 1G edge2-servprov-testlab ge-0/0/0 (UN) edge2 1G port - p2p services testing with callbacks

# create\_node

Retry Abort

node Juniper

Status	Current step	Customer	Started by	Started on	Last update	Related subscriptions
COMPLETED	Done	Default::Orchestrator-Core C...	SYSTEM	2/29/2024, 5:31:04 PM	2/29/2024, 5:39:49 PM	node edge1-mic-courtbrack1 ...



# Workflow Orchestrator

Workflow steps [Expand all](#)

- Start**  
success - 2/29/2024, 5:31:06 PM
- Construct Subscription model**  
success - 2/29/2024, 5:31:06 PM
- Create Process Sub**  
success - 2/29/2024
- Validate IP address**  
success - 2/29/2024
- Create node in Netbox**  
success - 2/29/2024

```

1  {
2    "payload": {
3      "name": "edge1-mic-courtbrack1"
4      "role": 11,
5      "site": 128,
6      "status": "active",
7      "tenant": "2",
8      "asset_tag": 30247,
9      "device_type": 14,
10     "primary_ip4": null,
11     "primary_ip6": null
12   },
13   "subscription": {
14     "node": {
15       "ims_id": 79
16     }
17   }
18 }
    
```

- Reserve or assign loopback addresses**  
success - 2/29/2024, 5:31:11 PM Duration 00:00:01
- Update node in Netbox**  
success - 2/29/2024, 5:31:13 PM
- Create DNS records**  
failed - 2/29/2024, 5:31:18 PM
- Create DNS records**  
failed - 2/29/2024, 5:31:18 PM
- Create DNS records**  
success - 2/29/2024, 5:39:48 PM Duration 00:08:30
- Provision node in NRM**  
success - 2/29/2024, 5:39:48 PM Duration 00:00:00
- Set subscription to 'active'**  
success - 2/29/2024, 5:39:48 PM Duration 00:00:00
- Unlock subscription**  
success - 2/29/2024, 5:39:48 PM Duration 00:00:00
- Cache Subscription and related subscriptions**  
skipped - 2/29/2024, 5:39:48 PM Duration 00:00:00
- Done**  
complete - 2/29/2024, 5:39:49 PM Duration 00:00:00

```

1  {
2    "class": "Exception",
3    "error": "Failed to get Zone ID for 49.44.87.in-addr.arpa.: zone not found"
4  }
    
```

```
@create_workflow("Create node", initial_input_form=initial_input_form_generator)
```

```
def create_node() -> StepList:
```

```
    return (
```

```
        begin
```

```
        >> construct_node_model
```

```
        >> store_process_subscription
```

```
        >> validate_ip_addresses
```

```
        >> create_node_in_ims
```

```
        >> reserve_loopback_addresses
```

```
        >> update_node_in_ims
```

```
        >> create_dns_records
```

```
        >> provision_node
```

```
    )
```

```
@step("Create node in Netbox")
```

```
def create_node_in_ims(subscription: NodeProvisioning) -> State:
```

```
    payload = build_payload(subscription.node, subscription)
```

```
    print(f"Payload: {payload}")
```

```
    subscription.node.ims_id = netbox.create(payload)
```

```
    return {"subscription": subscription, "payload": payload.dict()}
```

```
@step("Create DNS records")
```

```
def create_dns_records(subscription: NodeProvisioning) -> State:
```

```
    zone_name = 'nn.hea.net.'
```

```
    device = netbox.get_device(name=subscription.node.node_name)
```

```
    # netbox returns ranges rather than individual IPs
```

```
    # below converts returned strings to blocks, and then gets network address
```

```
    # https://docs.python.org/3/library/ipaddress.html
```

```
    ipv4_address = (ipaddress.ip_network(device.primary_ip4.address)).network_address
```

```
    ipv6_address = (ipaddress.ip_network(device.primary_ip6.address)).network_address
```

```
    six_connect.create_dns_record_set(zone_name,
```

```
                                     str(subscription.node.node_name)+"."+zone_name,
```

```
                                     str(ipaddress.IPv4Network(device.primary_ip4.address).network_address),
```

```
                                     str(ipaddress.IPv6Network(device.primary_ip6.address).network_address))
```

```
    return {"subscription": subscription}
```

+ New subscription

Start

Start /

# Welcome



Operator:

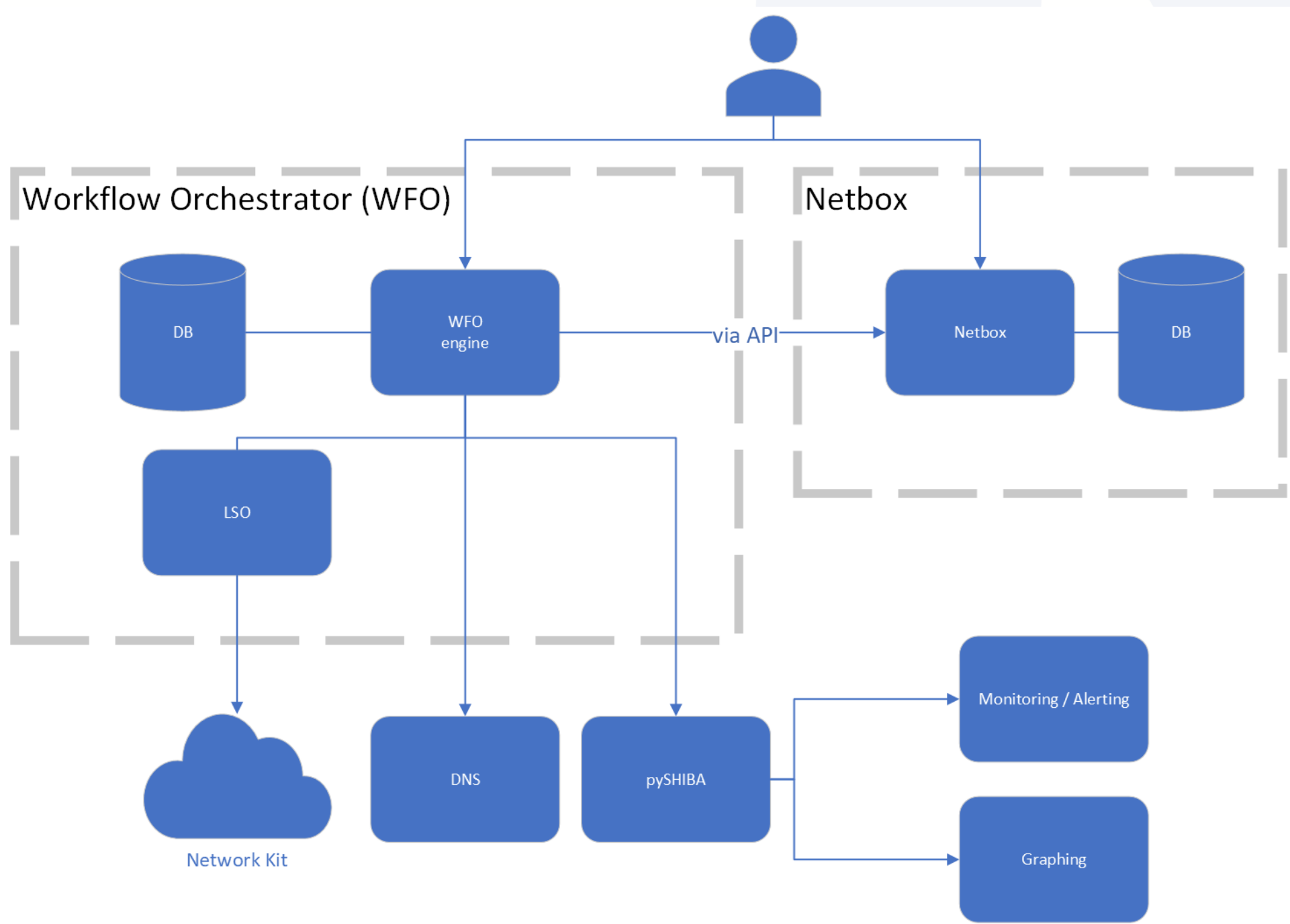
Clean Simple GUI Interface

Maintainers:

Multiple Connected Systems

CLI Interface

Python Coding



Processes involving multiple teams

Processes with a big time gap

Creating virtual resources

Creating user accounts

*Deleting* user accounts

*Validating* user accounts

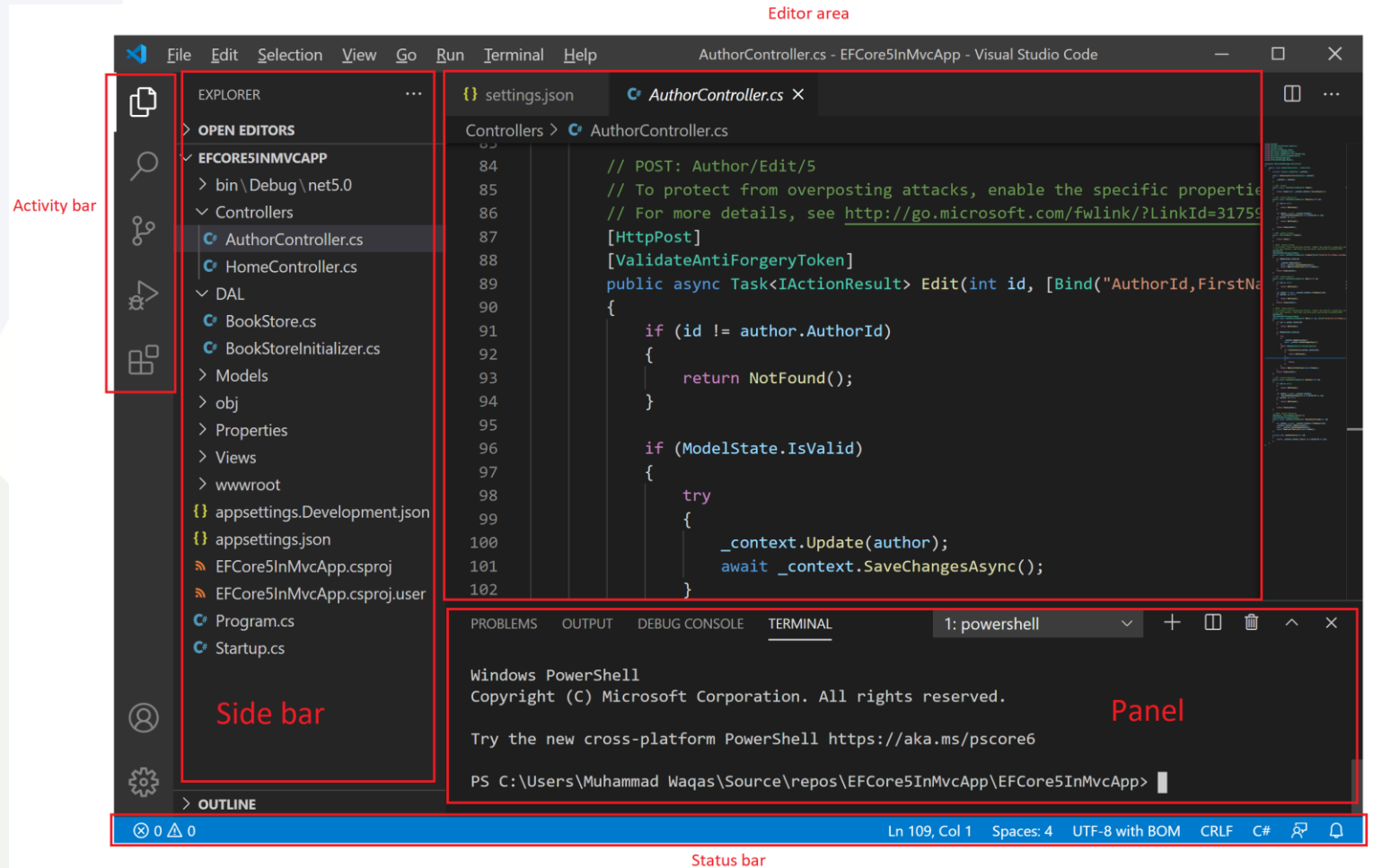


# Developer Tools

- Microsoft Visual Studio Code
- Gitlab
- Docker
  - Pending move to Kubernetes (K8S)

# Visual Studio Code

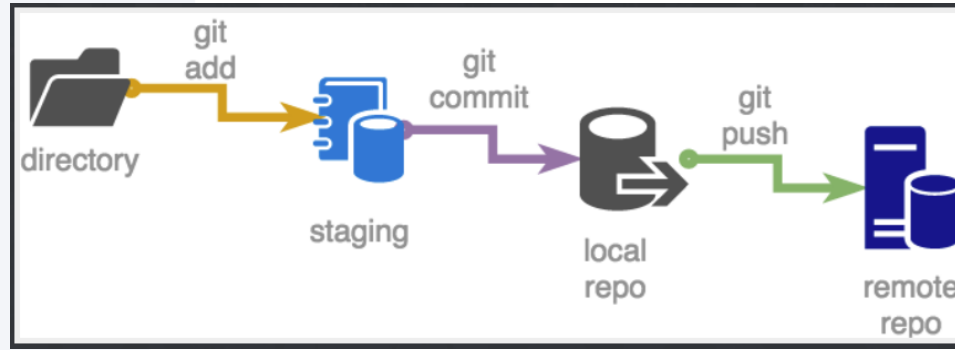
- Usual IDE features
- Git integration
- Remote Docker development





# Gitlab

- Repository Storage
- Version Control
- Collaborative Development
- CI/CD
- Issues & Kanban boards



**Merge request analytics**

Mean time to merge: 1 days

Throughput: The number of merge requests merged by month.

**Issue Boards**

Development

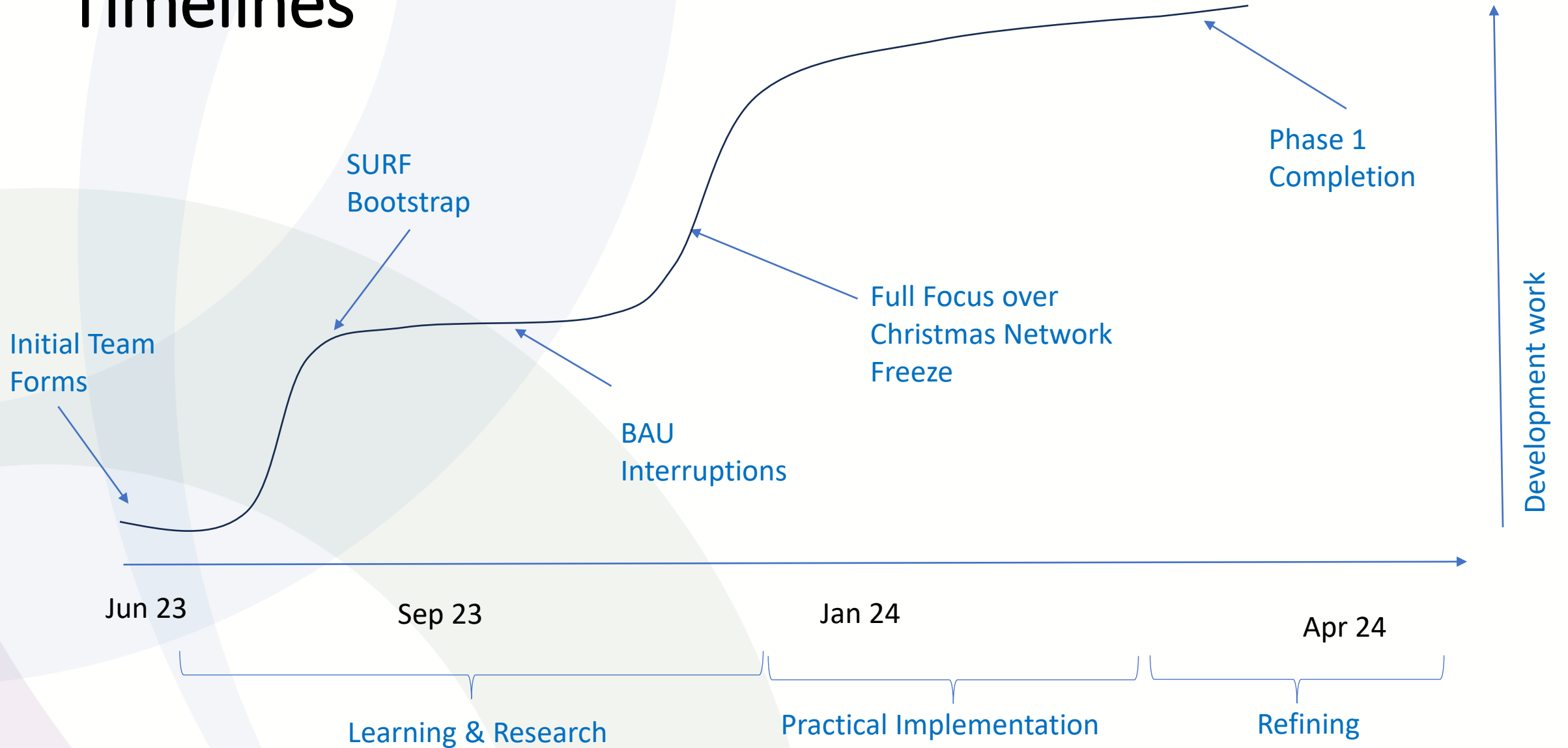
- Open (23)
- Phase 2 Start Sprint (10)
- Phase 1 wrap-up (7)
- Closed (68)

Commits	Pipelines	Line changes	Assignees
22		+10123 -2716	
4		+9 -1	
1		+75 -9	

# What could you do?

- Bash scripting
- AWS Lambda
- Gitlab CI jobs

# Timelines



# How do you start an Automation project

- Identify Needs / Define process
- Measure usefulness of Automation
  
- Build team
- Break down tasks / 2 Week Sprints
- Reach out to community
- Re-Use what you can
- Iterate and Refine



Questions ?

