

# One Network to rule them all

Netnod 2018, 14-15 Mar

Christian Adell

 @chadell0



**Networking** nowadays

# Challenges

- Scalability
- Containerisation
- Distributed Systems
- Multi-platform, Cloud
- High Performance applications
- Efficiency

**Traditional** network architectures/operations are not good enough

# Are we **ready**?

- Multi-vendor with **legacy** devices not well-suited for automation
- There is a lot of **new things to learn**
- Vendor **trainings** aren't (weren't) focused on this
- Automation **amplifies** everything (including mistakes)
- Usually, **not close to developers**, to the business

And most of the times, **we don't know where to start from...**

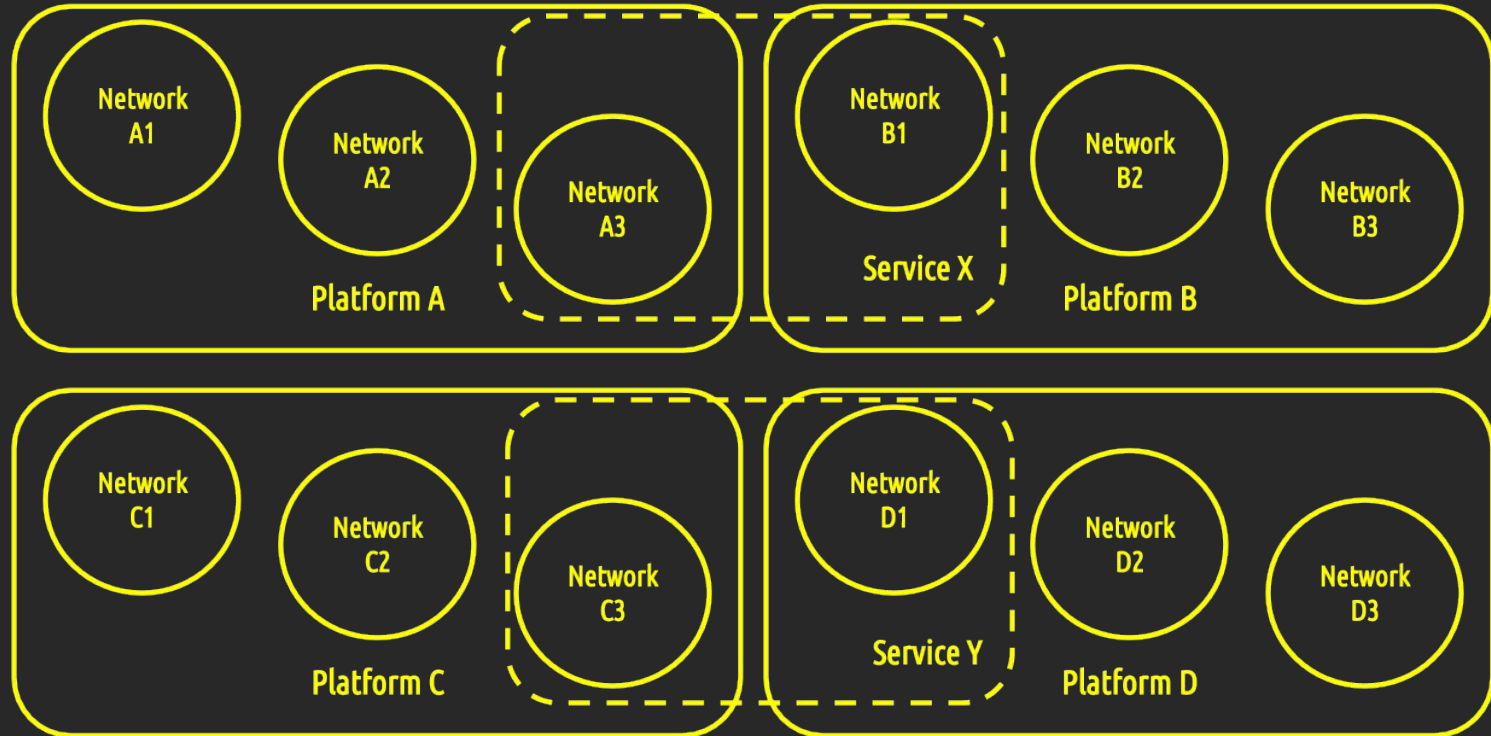
# How can we approach it?

- **APIs everywhere**, your network devices should support them
- Use **Data Models**, they will help you translate your will
- Take advantage of the **information** your network is providing
- Don't fear **dynamic infrastructure**
- Some **coding skills** will be needed
- Validate, validate and **validate** again

Start by solving simple problems... keeping **applications** on your focus

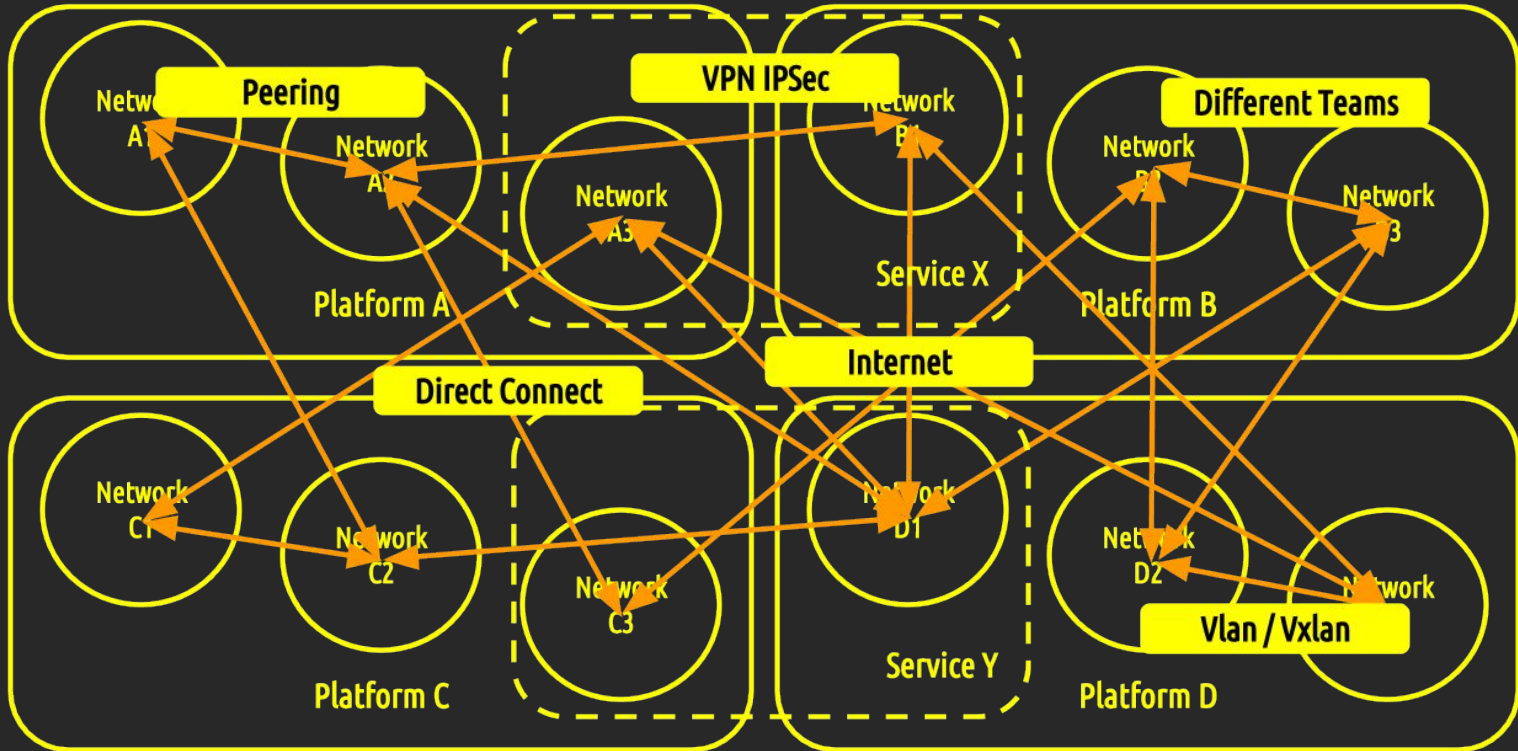
**A brief story of a **network** service**

# Typical IT ecosystem





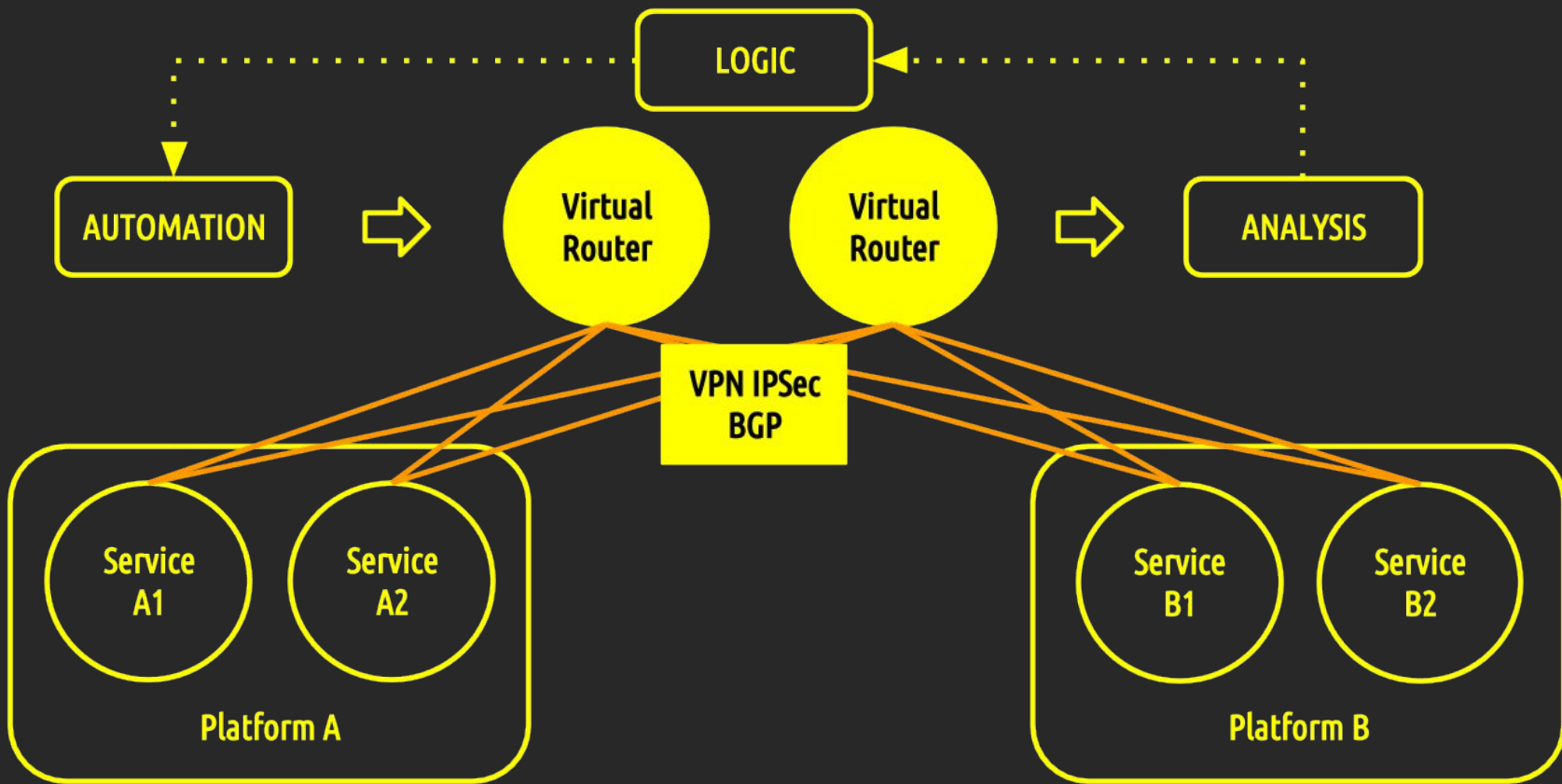
# How the network looks like



# Downsides

- By default, inter/intra platform communications use **Internet** which is not (always) the most **performant, secure and cheapest** communication channel
- Manual network provisioning doesn't work in terms of **speed and reliability**
- Prone to **errors** and lack of consistency
- Some communications still need **network layer security** (no TLS)

**We tried to solve all in one**

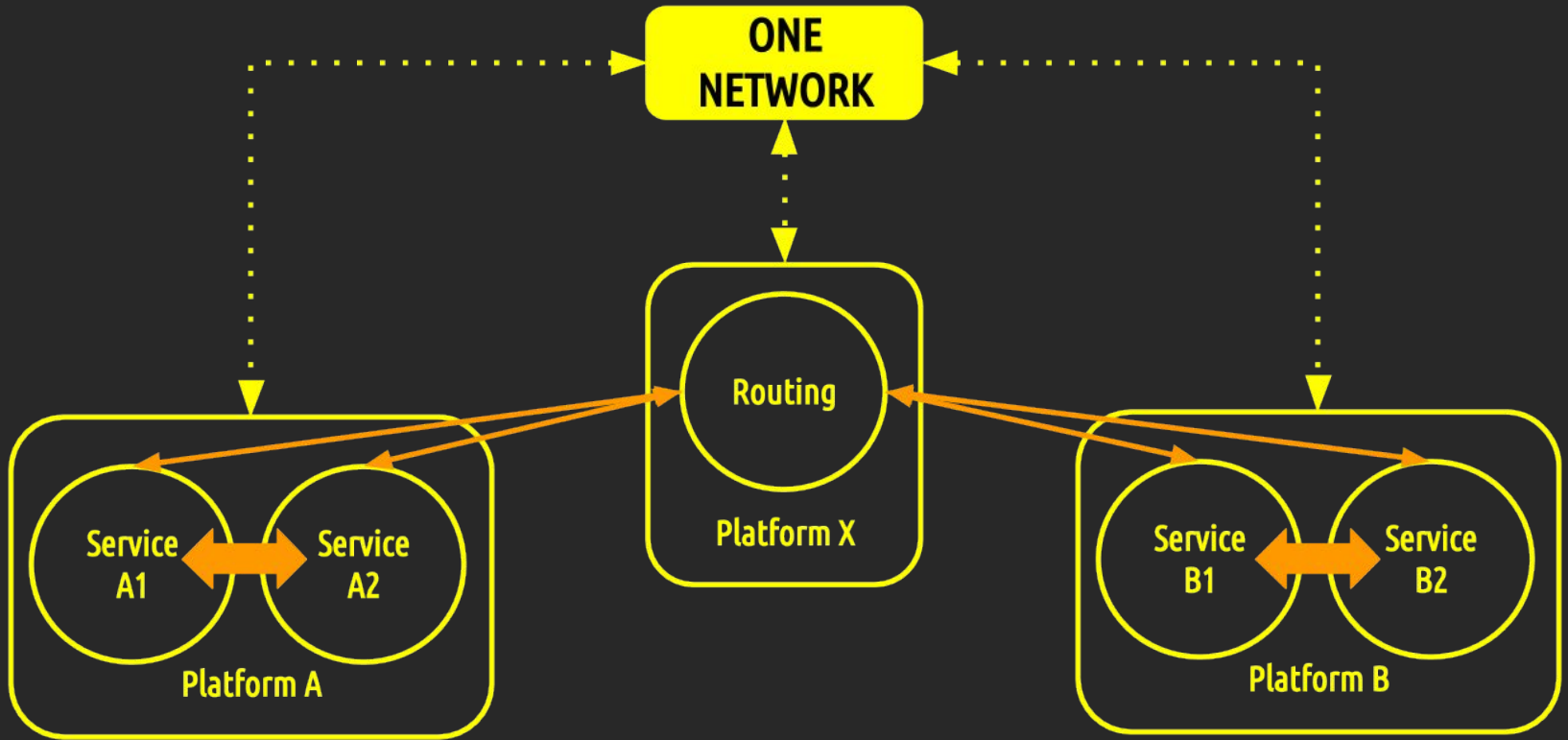


... and we **failed**

# (non-technical) Lessons learned

- Think as your **users** will do
- Get **feedback** as soon as possible, iterate!
- The solution should **flexible** enough to accommodate several underlying solutions
- Evaluate **current** needs case-per-case (capability, performance, cost, etc.)
- Apply Pareto Rule, **focus** on solving most urgent needs first

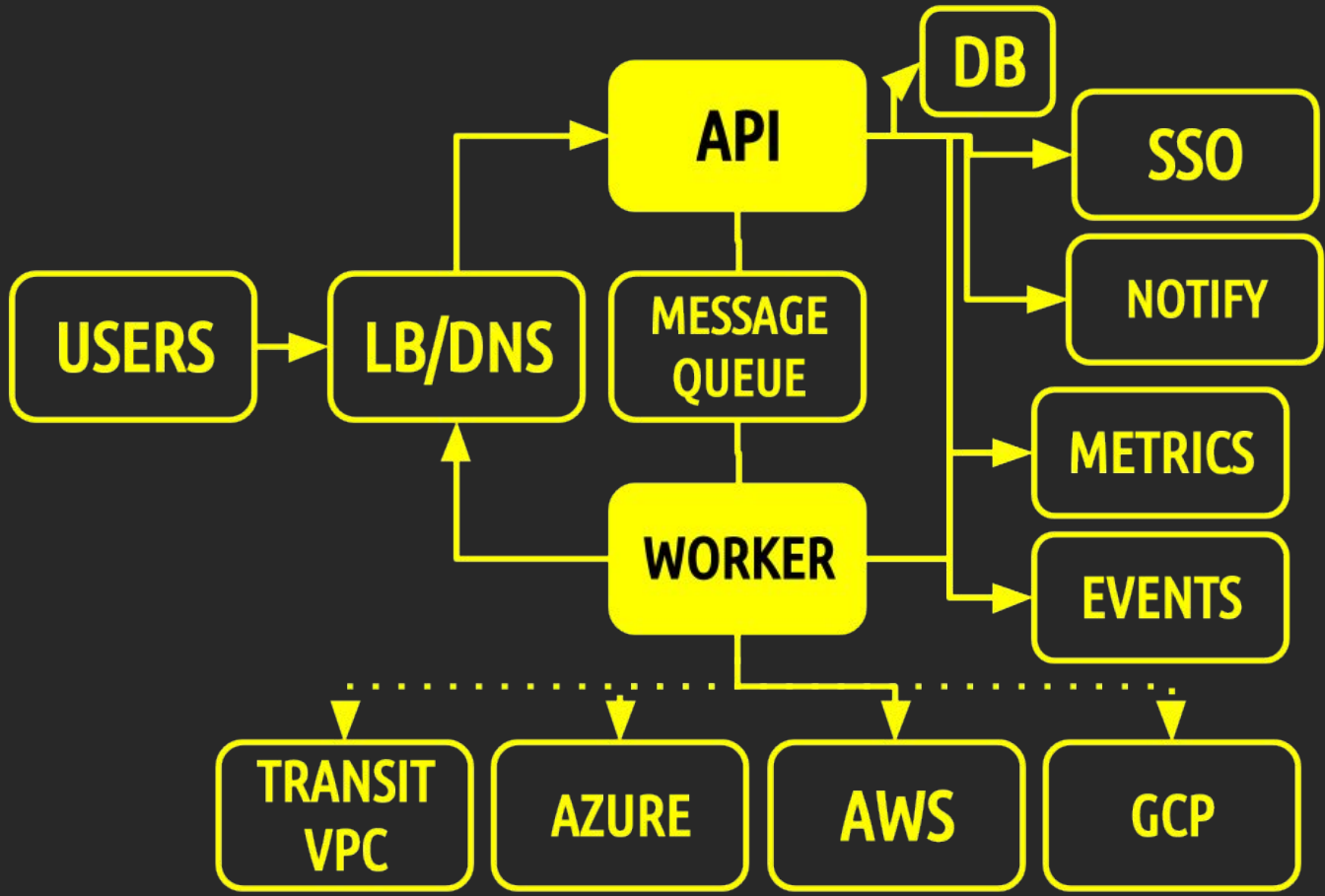
Then, we created a **network service**

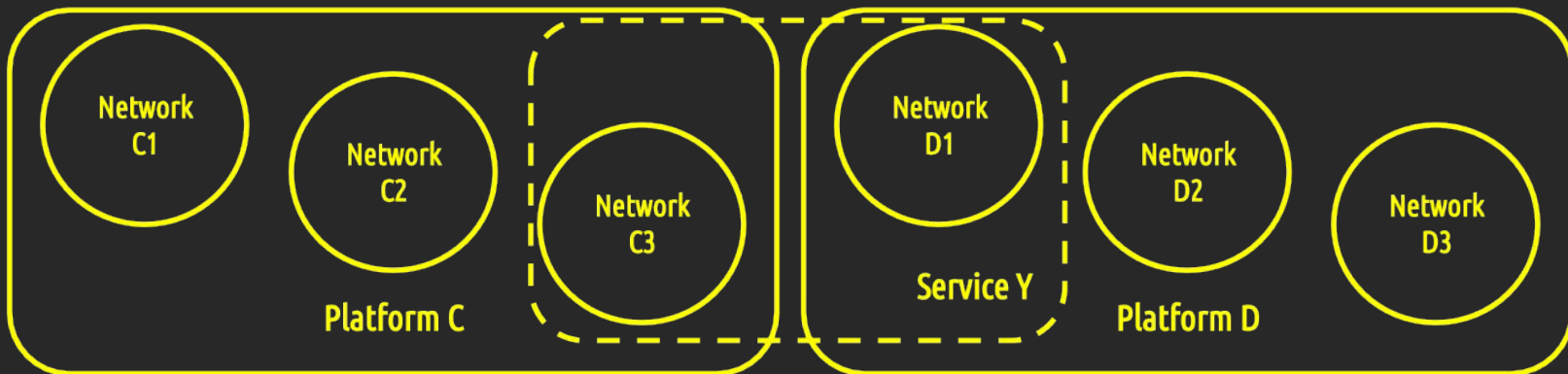
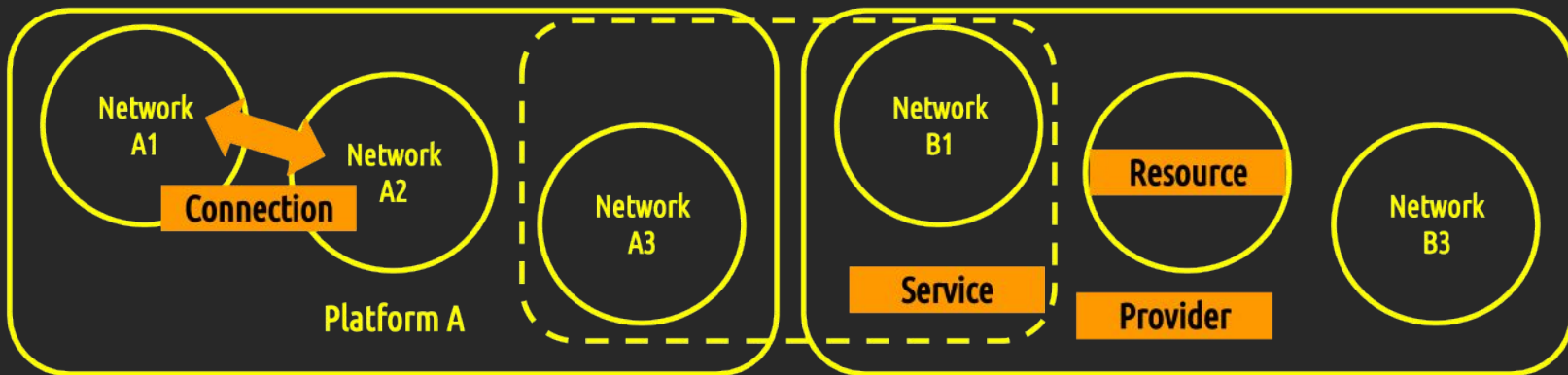




# Requirements

- Easy onboarding / **self serve**
- Users should be **autonomous** to handle connections
- **Abstract** all network details from users and pick the **best** option in every case
- Support **several** providers/platforms
- Offer a **secure** service
- Continuous monitoring of connection **status**





## SERVICE

```
{
  "owners": [
    "u'1ed3aef1-3e4d-4372-95e1-0890a1148a48'"
  ],
  "name": "u'Dummy Service Name",
  "providers": [
    "u'6445dbee-3eaa-4911-b387-1ee79805f75e'"
  ],
  "id": "u'd44ff67c-9ebf-4a96-b9f0-55336860f6b6'",
  "resources": [
    "u'14b586c7-7b2a-4469-ba11-743a0d7ce219'"
  ],
  "description": "u'Dummy service for testing purposes'"
}
```

## PROVIDER

```
{
  "account": "u'1111111111'",
  "name": "u'Dummy Provider Name",
  "service": "u'd44ff67c-9ebf-4a96-b9f0-55336860f6b6'",
  "region": "u'ap-southeast-1'",
  "id": "u'6445dbee-3eaa-4911-b387-1ee79805f75e'",
  "provider_type": "u'AWS'"
}
```

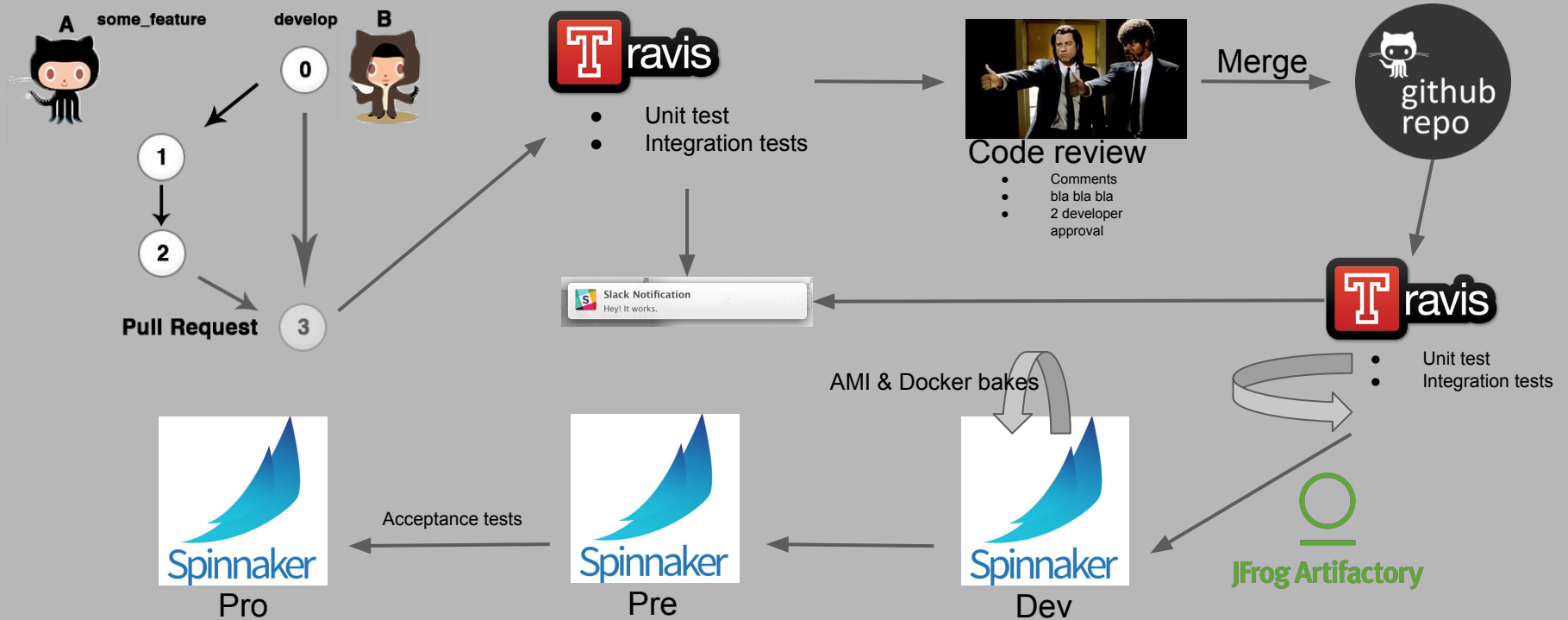
## CONNECTION

```
{
  "state": "u'ESTABLISHED'",
  "name": "u'Dummy Connection Name",
  "desired_state": "u'ESTABLISHED'",
  "id": "u'e975ecfd-9bd6-4ea3-ab56-6193bb752cba'",
  "resource_left": "u'14b586c7-7b2a-4469-ba11-743a0d7ce219'",
  "details": {
    "peering_id": "u'pcx-1a1a1a1a'"
  },
  "resource_right": "u'342ac6d7-74fd-4290-a52c-fbdd325b95ef'",
  "ctype": "u'AWS_PEERING'"
}
```

## RESOURCE

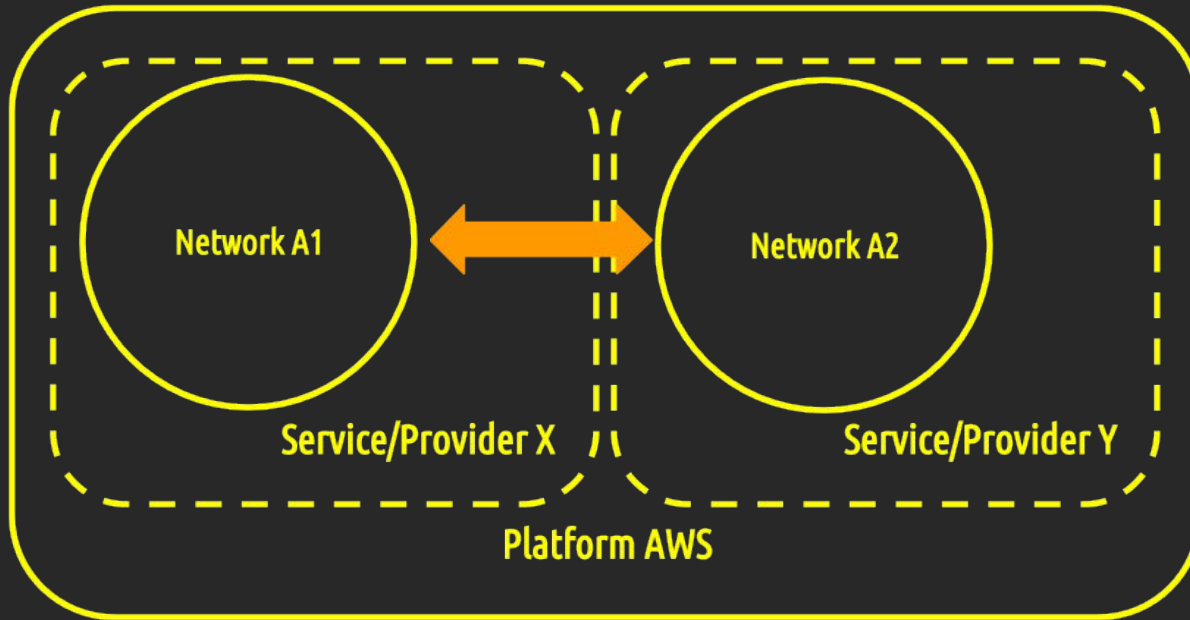
```
{
  "subnet": "u'10.69.249.192/28'",
  "name": "u'AWS\\vpc-1a1a1a1a'",
  "service": "u'd44ff67c-9ebf-4a96-b9f0-55336860f6b6'",
  "l4filters": [
  ],
  "provider": "u'6445dbee-3eaa-4911-b387-1ee79805f75e'",
  "id": "u'14b586c7-7b2a-4469-ba11-743a0d7ce219'"
}
```

# Continuous Integration & Deployment



**Demo**

# Scenario



```
...
Create GSN assets
  Service gsn-test-service-2018021019581518289097-profile1 created with ID: 1d410f61-bf0d-41f9-b656-46847350fdce
  Provider gsn-test-provider-2018021019581518289097-profile1 created with ID: 95d87197-8ec8-492f-b0e2-7364fd6965d6
  Service gsn-test-service-2018021019581518289097-profile2 created with ID: 12a8f1a6-e843-48c2-9a85-bbc02c0a6975
  Provider gsn-test-provider-2018021019581518289097-profile2 created with ID: 001e1478-fedb-44f7-8e67-29958e449e6b
  OK
...
Wait for GSN to discover resources
  Resource 48f9fce1-348d-470c-bf60-c7668cf498e2 discovered in Provider 95d87197-8ec8-492f-b0e2-7364fd6965d6
  Resource cb3ec005-585b-42d8-83a8-b932bcd13b52 discovered in Provider 001e1478-fedb-44f7-8e67-29958e449e6b
  OK
...
Create GSN connection
  Connection between resources 48f9fce1-348d-470c-bf60-c7668cf498e2 and cb3ec005-585b-42d8-83a8-b932bcd13b52 has been created
  Connection between resources 48f9fce1-348d-470c-bf60-c7668cf498e2 and cb3ec005-585b-42d8-83a8-b932bcd13b52 has been created
  OK
...
Wait for the GSN connection to become ESTABLISHED  OK
...
Check instances can ping each other now  OK
...

```

```
64 bytes from 10.69.251.52: icmp_seq=96 ttl=255 time=0.591 ms
64 bytes from 10.69.251.52: icmp_seq=97 ttl=255 time=0.571 ms
64 bytes from 10.69.251.52: icmp_seq=98 ttl=255 time=0.539 ms
64 bytes from 10.69.251.52: icmp_seq=99 ttl=255 time=0.537 ms
64 bytes from 10.69.251.52: icmp_seq=100 ttl=255 time=0.548 ms
64 bytes from 10.69.251.52: icmp_seq=101 ttl=255 time=0.464 ms
64 bytes from 10.69.251.52: icmp_seq=102 ttl=255 time=0.501 ms
64 bytes from 10.69.251.52: icmp_seq=103 ttl=255 time=0.437 ms
```



# Why our developers use it?

- They don't care about **underlying** network details
- They always use the **best** possible network solution
- They have **one API** to handle everything
- They are **notified** about connections' health
- They need an out-of-the-box **multiple platform** connectivity
- They get **visibility** about network dependencies

# Takeaways

- **Don't be afraid** of going out of your comfort zone
- Learning coding will give you **superpowers**
- At some point, you will need to **join** pieces
- Adopting a DevOps approach will **speed up** your business (and career)
- **Networking** is a key skill in IT, bring it **close to the business**

**Thanks** for your attention

# Related material

Some **learning** stuff: <https://github.com/chadell/learning>

**Q/A**

