



INFRAHUB

Unified data management for
infrastructure automation and AI

The screenshot displays the InfraHub web application interface. On the left is a sidebar with a navigation menu. The main content area shows a 'Device' page with a table of network devices. The table has columns for Device, Site, Name, Description, and Type. The devices listed include core, edge, and leaf nodes for two different sites (atl1 and den1).

INFRAHUB

Search 🔍

Organization
Location
Device Management
Circuit Management
Network Configuration
Routing & Peering
Services
IPAM
Other

Proposed Changes
Branches
Object Management

Device 30 🔄
Generic Device object

Search Device 🔍

Device	Site	Name	Description	Type
atl1-core1	atl1	atl1-core1	-	MX204
atl1-core2	atl1	atl1-core2	-	MX204
atl1-edge1	atl1	atl1-edge1	-	7280R3
atl1-edge2	atl1	atl1-edge2	-	7280R3
atl1-leaf1	atl1	atl1-leaf1	-	7010TX-48
atl1-leaf2	atl1	atl1-leaf2	-	7010TX-48
den1-core1	den1	den1-core1	-	MX204
den1-core2	den1	den1-core2	-	MX204
den1-edge1	den1	den1-edge1	-	7280R3
den1-edge2	den1	den1-edge2	-	7280R3

About me

- Previously at Cloud Temple and Veepee
- Based in Amsterdam (NL) since 2017
- Joined OpsMill in 2023
- Swiss Army knife





Experienced Leadership

Founded by **Damien Garros** and **Raphael Maunier**

Strong Backing

Proudly supported by leading investors:

**Serena Capital, Partech, and OVNI
Capital**

Expert Team

15+ **infrastructure automation
specialists** across Europe and North
America drives our vision forward.

Trusted by Industry Leaders

Major Financial Institution: \$1M+ investment, billion-dollar security requirements

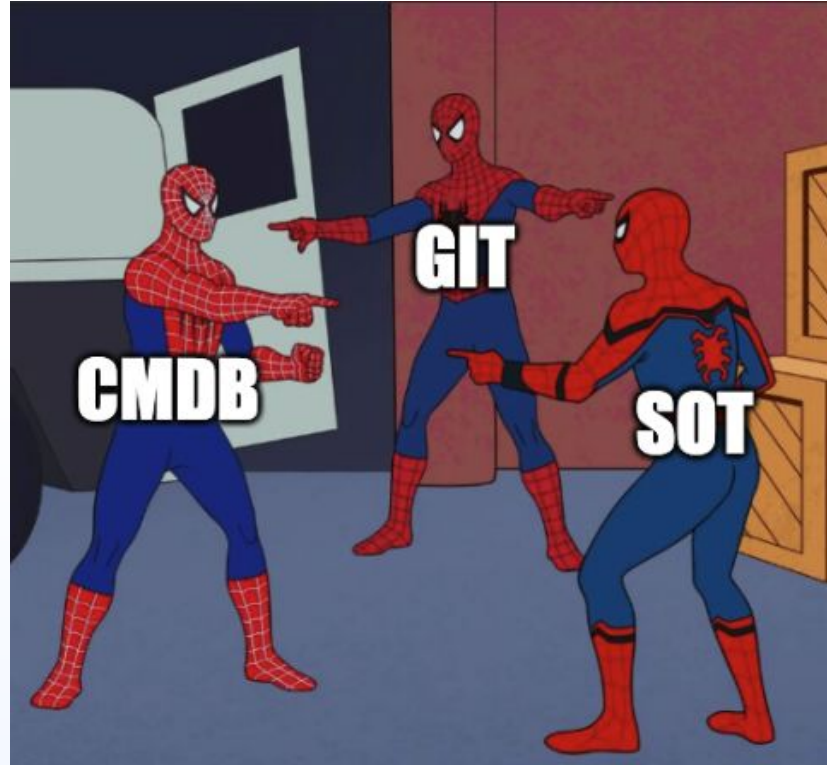
Global Media Brand: 2,000-5,000 device environment

EU Service Provider: ISO 27001 certified, security audit compliance

Hedge Fund: Custom RBAC and compliance workflows



Fragmented Truth → Fragile Automation

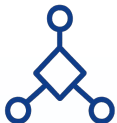


Current solutions can't cope with this data complexity



CMDB

Good for asset tracking.
Not useful for automation.
Not built for DevOps.



IaC/Git

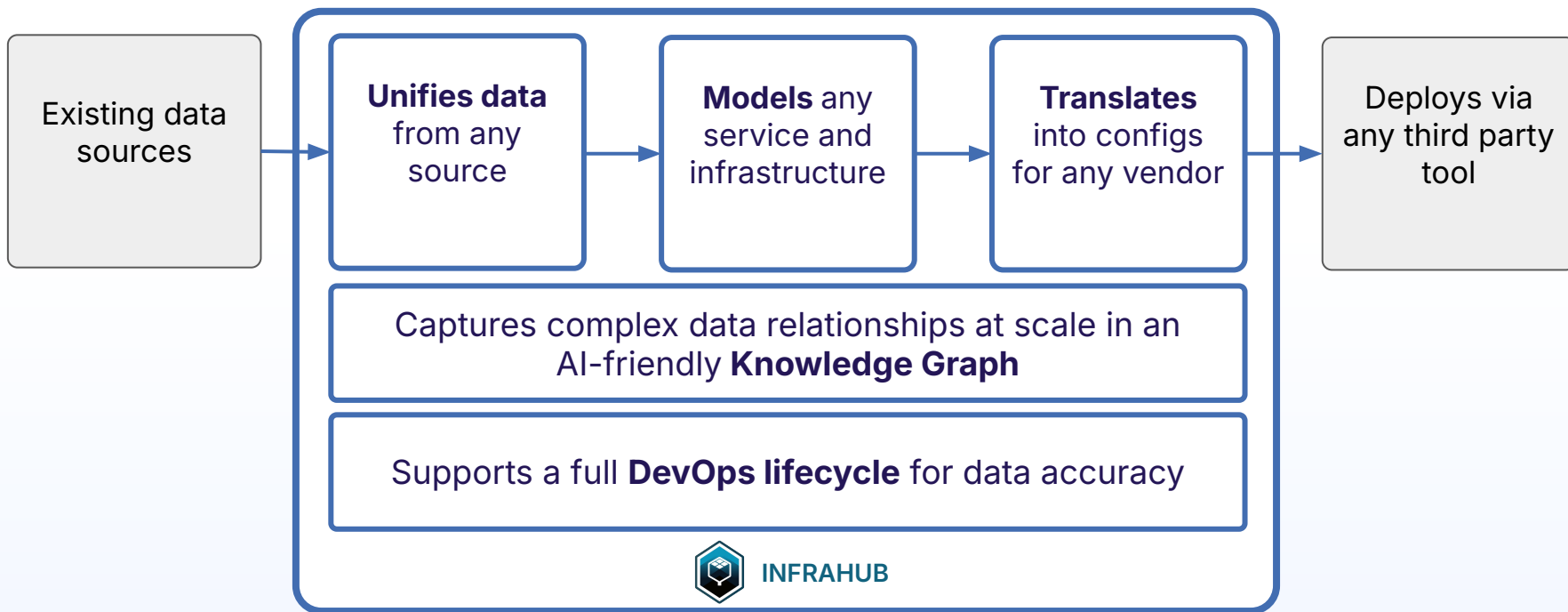
Good for cloud API automation.
Strong DevOps support.
Data model too simple for infra.



Infra Mgmt Tools NSoT

Inflexible or vendor-locked.
Poor DevOps support.

Infrahub: Purpose-built data management platform



Features you'll actually use

Security & Compliance

RBAC

Metadata

Abstraction

Template

Profile

CI Pipeline

Proposed Changes

User Checks

Tests
(Unit & Integration)

Idempotency

Resource Manager

Generator

Config Generation
(Artifact)

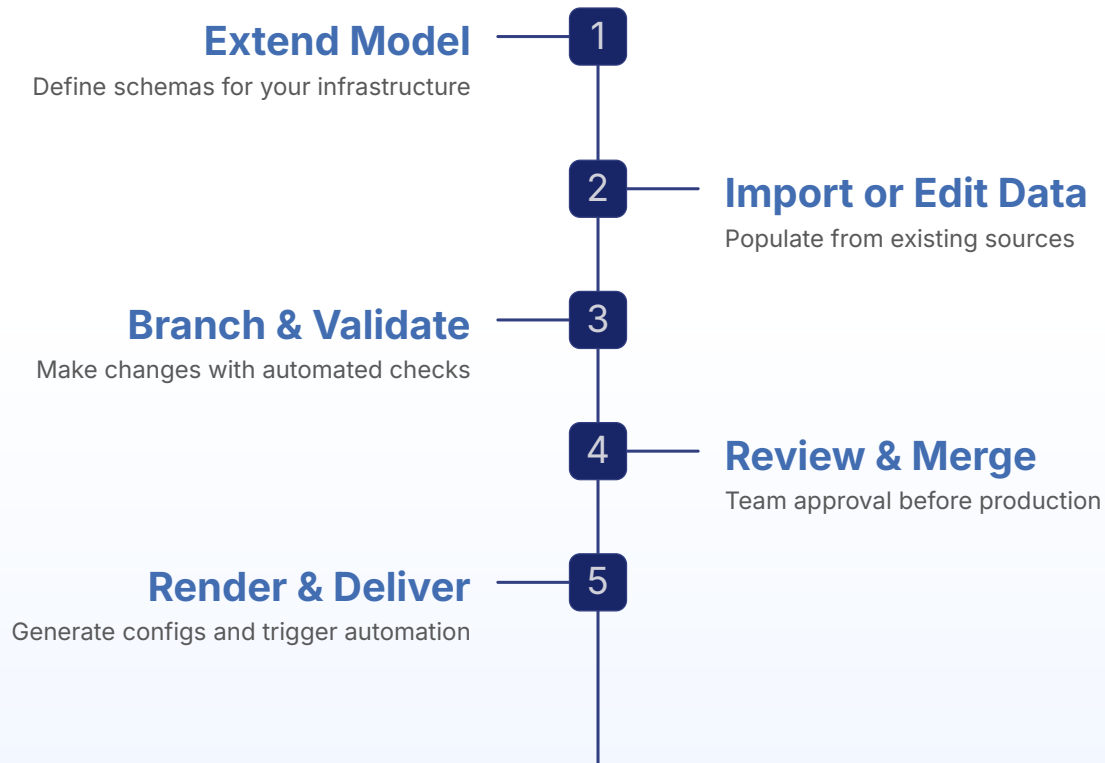
Git Integration

Version Control

GraphQL

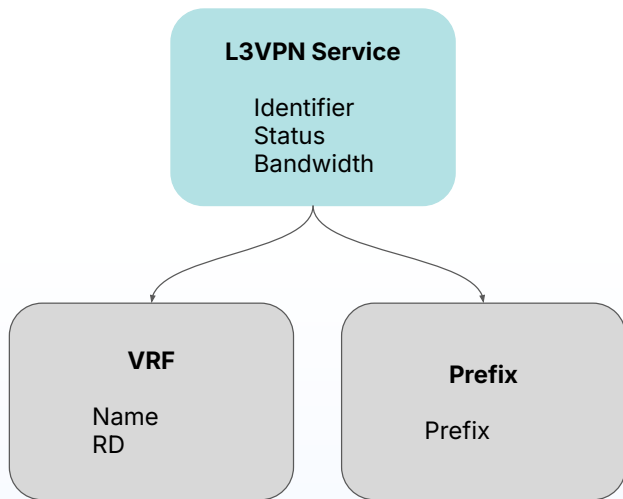
Flexible Schema

From model to automation



Schema Definition

Example: L3VPN Service Definition



Build a robust data model that can be leveraged across your entire pipeline

! example.yml

```
1  nodes:
2    - name: L3VPN
3      namespace: Service
4      icon: mdi:ethernet
5      attributes:
6        - name: identifier
7          kind: Text
8          unique: true
9          optional: false
10       - name: status
11         kind: Dropdown
12         optional: false
13         default_value: draft
14         choices:
15           - name: draft
16           - name: in-delivery
17           - name: active
18           - name: in-decomissioning
19           - name: decomissioned
20       relationships:
21         - name: vrf
22           peer: NetworkVRF
23           cardinality: one
24           optional: true
```

Integrate various data source

Customer data synced from ServiceNow automatically populates the service model



Create strong integration and a comprehensive overview of your network implementation.

From Tribal knowledge to code

| “Network design lives in Visio diagrams and senior engineers' heads”

Infrahub Generators are python scripts that **encode your design patterns** and automatically **translate service requests into implementation**.

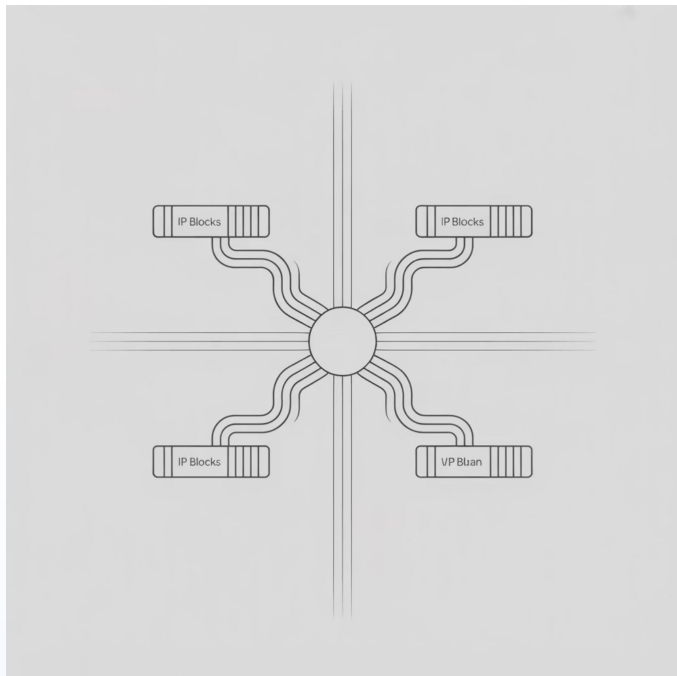


VRF should be named after service identifier

IP address should be allocated from the designated customer prefix

Gateway IP address is the first available IP of the prefix

Intelligent Resource Management



Define Resource Pools

IP prefixes, VLAN ranges, AS numbers, and other network resources

Automatic Allocation

Works with generators, no more spreadsheet tracking or manual assignment

Guaranteed Uniqueness

Prevent conflicts before they happen with built-in validation

Result: Ship consistent, predictable services with engineering confidence

From Data Model to Device Reality

Transforms let you **define vendor-specific logic** → Converts a generic service into a deployable configuration for the field.



This approach maintains a **clear separation between your service definition and the technical implementation** details

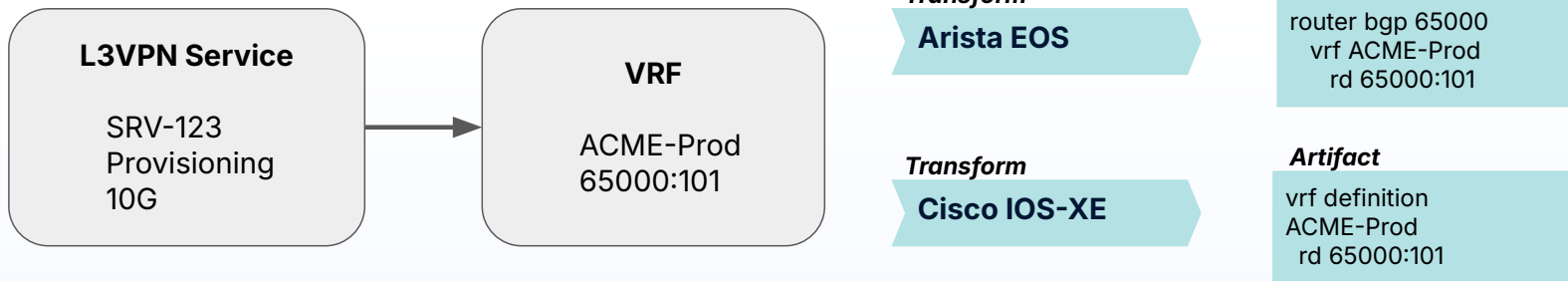
Vendor Abstraction in Practice

Mixed Technology Infrastructure

Same service spanning Cisco on Site A and Arista on Site B

Cross-Platform Services

Cisco routing with Palo Alto firewalls



Vendor complexity is contained in transforms – your service logic remains clean and portable

Version Control & Collaboration

All changes can be done in an isolated branch

Multiple branches can coexist at the same time.

All changes from a branch are being tracked and can be reviewed / approved before integrating them in the main branch

The image displays two screenshots of the OpsMill interface, illustrating version control and collaboration features.

Top Screenshot: Update IPs in JFK1

The top screenshot shows the "Update IPs in JFK1" interface. The left sidebar lists navigation options: Organization, Location, Device Management, Circuit Management, Network Configuration, Routing & Peering, Services, Proposed Changes, and Object Management. The main content area shows the "Update IPs in JFK1" interface, indicating that "Admin wants to merge jfk1-update-edge-ips into main". The interface includes tabs for Overview, Data, Files, Artifacts, Schema, Checks (0), and Tasks (1). A summary bar shows "Updated 1 minute ago" and buttons for "Refresh diff" and "Rebase". The "Data" tab displays a tree view of the configuration, showing the "jfk1-update-edge-ips" branch being merged into "main". The tree view shows the following structure:

- Device
 - jfk1-edge2
 - interfaces
 - Ethernet1
 - jfk1-edge1
 - IP Address
 - 10.1.0.32/31
 - 10.1.0.33/31

The "Data" tab also shows a comparison of the "main" branch and the "jfk1-update-edge-ips" branch, highlighting the changes in the "Ip Addresses" section.

Bottom Screenshot: Proposed changes

The bottom screenshot shows the "Proposed changes" interface. The left sidebar is the same as the top screenshot. The main content area shows the "Proposed changes" interface, indicating that there are 2 proposed changes. The interface includes a search bar and buttons for "Opened 2", "Closed 0", and "+ New proposed change". The "Data" tab displays a table of proposed changes:

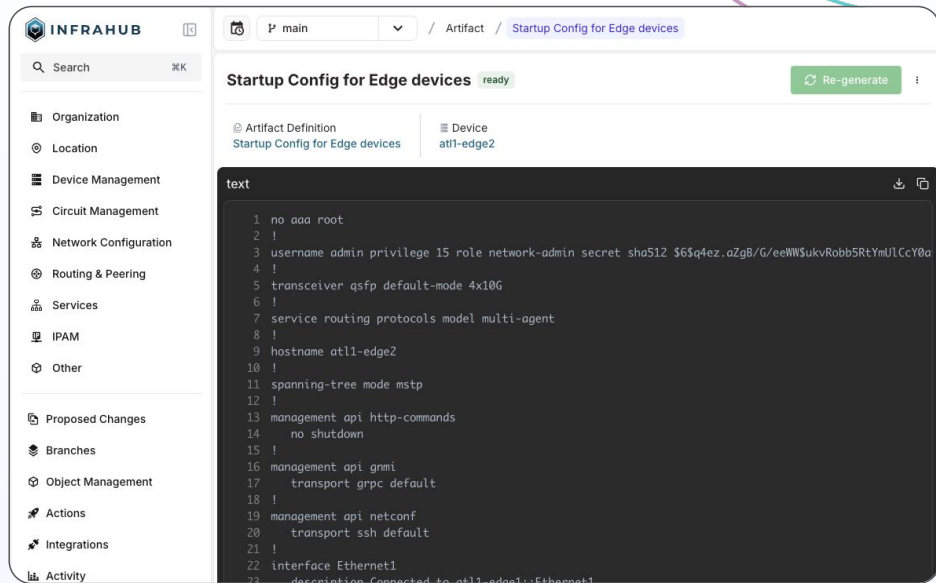
Name	Data	Checks	Tasks	Artifacts	Reviewers
PC2	branch2	5	12	0	No reviewers
Update IPs in JFK1	jfk1-update-edge-ips	1	8	0	

Artifact / Config generation

Artifacts of all kinds are supported:

- Text file / Configuration file
- JSON data
- XML
- Markdown
- YAML
- SVG
- CSV

Artifacts are immutable and all versions of an artifact will remain available.



Deployment Integration

Infrahub adopts an API-first approach and all features in the frontend are available via our GraphQL or REST APIs.

OpsMill provides SDKs (Python, GO) and libraries to facilitate integration with ecosystem automation tools.

Ansible
Terraform
Nornir

Orchestrator

Infrahub Sync

Python / GO SDK

GraphQL API

Rest API



Infrahub OSS

<https://github.com/opsmill/infrahub>

