# Ansible Automation Platform - Team Onboarding

## Or the demo from Hell

Toni Schmidbauer

Senior Architect

# What we'll discuss today

▶ Introduction / Scenario

▶ Onboarding goals

▶ Demo

▶ Open Issues / Learning

# Introduction / Scenario



▶ So you have AAP running (lucky? you) and the first automation jobs are working great

▶ You tell everybody how cool Ansible and AAP are and how much time for other stuff you now have (Fallout S2 anyone?)

▶ Suddenly another team/department also want to use APP (shite)

# Introduction / Scenario

▶ You could say no, this is my Platform

# Introduction / Scenario

But because you are great and very helpful  and using Ansible/AAP only for one team is a waste to be honest, you

▶ create a Team, or another Organization within AAP

▶ create users, tokens,  grant access

▶ create GIT repos, credentials

▶ help the Team creating their first job template, inventory
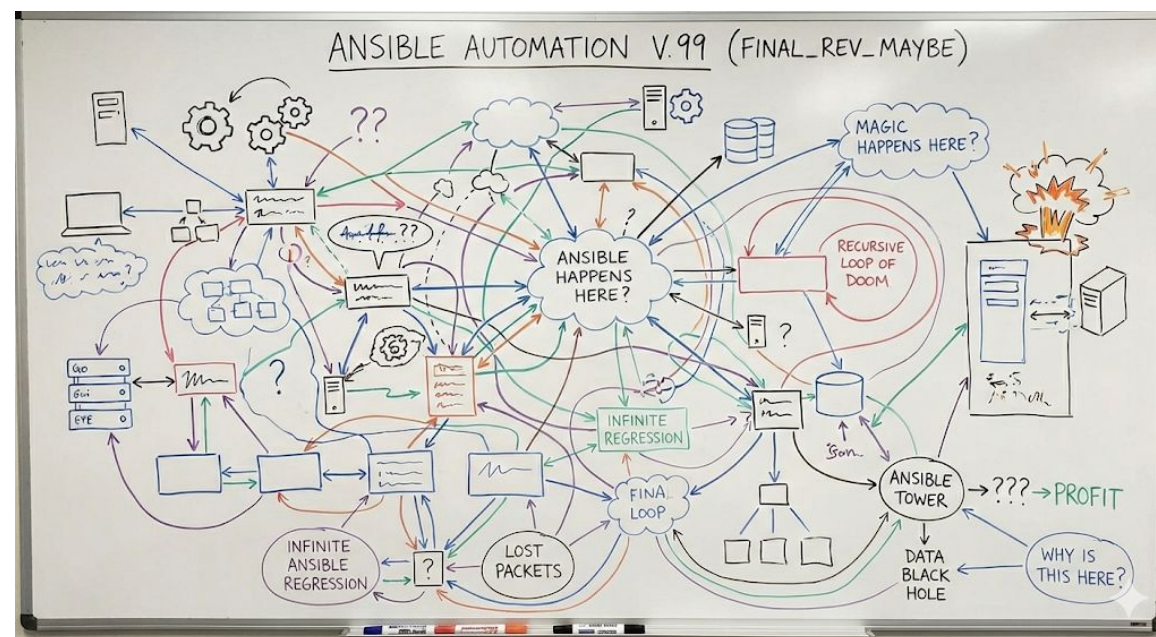
▶ and ….

You did a great job, the other team is happy!

But now the next teams comes along (shite)

# So it's time to automate the hell out of that stuff

# The platform team

- We have a Team that operates AAP – aka the **Platform team**

  - Most of the time they are also heavy users of AAP

- We have **other departments** that would like to use AAP independently

- These departments are our customers

- The happier our customers, the happier the platform team

- https://platformengineering.org/

**PLATFORM STRATEGY**

INNOVATION THROUGH HARMONIZATION

Gregor Hohpe

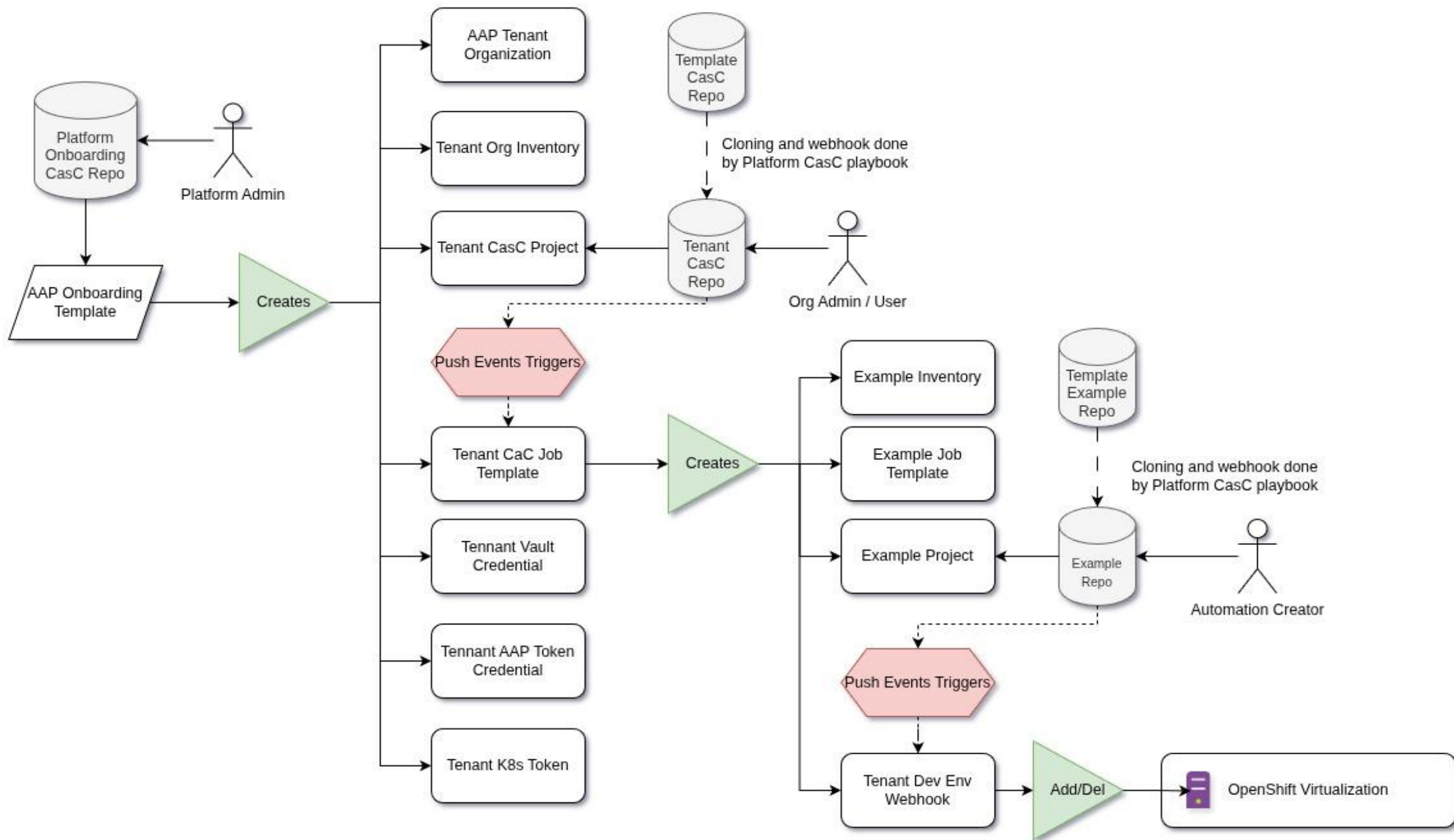**An Architect Elevator Guide**

With contributions by Michele Danieli
and Jean-Francois Landreau

# Onboarding goals

- ❏ Onboarding should be automated

- ❏ Teams should be able to / are forced to use Configuration as Code

- ❏ Teams should have an example repository with automation code, aka a golden path

- ❏ We should be able to enforce naming conventions

- ❏ Teams should get the ability to deploy test environments

AAP Tenant Organization

Template CasC Repo

Tenant Org Inventory

Platform Onboarding CasC Repo

Platform Admin

Cloning and webhook done by Platform CasC playbook

Tenant CasC Project

Tenant CasC Repo

Org Admin / User

AAP Onboarding Template

Creates

Push Events Triggers

Tenant CaC Job Template

Creates

Example Inventory

Template Example Repo

Example Job Template

Cloning and webhook done by Platform CasC playbook

Tennant Vault Credential

Example Project

Example Repo

Automation Creator

Tennant AAP Token Credential

Push Events Triggers

Tenant K8s Token

Tenant Dev Env Webhook

Add/Del

OpenShift Virtualization

10

# DEMO

# Onboarding goals

✓ We automated everything

✓ Teams get a CaC repository (*tenant-org-config*)

✓ Teams get a golden template of automation code (*tenant-example*)

✓ We use Policy as Code to enforce naming conventions

✓ We create development environments based on feature branches in the golden path example repo

# Read the source Luke

✓ https://github.com/tosmi-ansible/aap-onboarding

✓ https://github.com/tosmi-ansible/template-org-config

✓ https://github.com/tosmi-ansible/template-example-project

✓ https://github.com/redhat-cop/infra.aap_configuration

# Learnings

# Learnings

▸ *infra.aap_configuration.dispatch* is used in the tenant CasC code repository

· Using this for tenant onboarding requires complicated variables merges

▸ The *onboarding.yaml* playbook uses *ansible.platform, ansible.controller …* directly

▸ *ansible-navigator* is your friend, most of the time

15

# Learnings

▸ There a lot of moving wheels, wrapping your (my) head around everything is difficult. **Documentation is key.** (*Note: 2026-02-16 22:25: I have none(yet)*)

▸ Use an OAuth token for API Access (Users/API Tokens)

· e.g. controller_api (to query api objects) does not work otherwise

▸ GitHub Webhook authentication does not use bearer token, the secret is something else

· https://docs.github.com/en/webhooks/using-webhooks/validating-webhook-deliveries

16

# Learnings

▶ *ansible.platform.token* sets *aap_token*

- and wiped my existing admin *aap_token*

- yes, that is mentioned in the *ansible.platform.token* docs, thank you for nothing!

▶ When building multi-arch container images, wipe everything before a new build. Tags were pointing to old images.

▶ *ansible.controller* currently does not support setting policies, need to use *uri*

# Learning OPA / Rego

‣ Rego is weird at the beginning but learning the basics took me 2-3h

‣ [Styra getting started course](#)

‣ https://play.openpolicyagent.org/

‣ https://regex101.com (OPA uses RE2)

‣ And it's pretty cool, e.g. you could host our policy bundle centrally

# AAP Policy as Code

▸ Policies are assigned to Orgs / Job Templates / Inventories

▸ Policies are enforced when a job template is triggered

▸ [Policy inputs](#)

▸ Expected output

```
{
  "allowed": false,
  "violations": [
    "No job execution is allowed",
    ...
  ],
  ...
}
```

# Open Issues / Improvements

# Open issues / improvements

- *https://github.com/tosmi-ansible/aap-onboarding/issues*
- Offboarding improvements (iterate over objects)
- Static node ports for VM's (because –ENOTIME)
- K8s authentication is a hack
  - ❏    create ClusterRole, ServiceAccount and Token
- Hub was out of scope
- Documentation improvements
- . . .

# Thank you

Red Hat is the world's leading provider of enterprise open source software solutions. Award-winning support, training, and consulting services make Red Hat a trusted adviser to the Fortune 500.

linkedin.com/company/red-hat

youtube.com/@redhat

facebook.com/RedHat

x.com/RedHat

Red Hat