



# How LinkedIn performs maintenances at Scale

Akash Vacher

13 Oct 2021



# Agenda

- 1 Motivation
- 2 Revisit SRE principles
- 3 The solution
- 4 The road ahead



Motivation

# What is a maintenance?

A planned activity on the underlying IT infrastructure for the purpose of hardware or software upkeep

# The Problem

- Infrastructure maintenances can cause real impact
- Headroom check for each service is required right before the maintenance
- Some services require non-trivial headroom checks

# The Problem

- More hardware, more maintenances
- Servers across multiple datacenters
- Single ToR network switches



# Revisiting SRE principles



## Embracing Risk

- Error budgets quantify acceptable unreliability
- Applied to a service dimension: availability, latency etc.
- Maintenances gone wrong can eat into the error budget





## Eliminating Toil

- Work that is manual, repetitive, devoid of enduring value
- Scales linearly as the service grows
- Significant human time spent for safety during maintenances



# Automation

- Problems of scale
- Consistency with faster operations
- Automatic systems provide a platform that can be extended



The solution

# The solution

- **STORU**: Single ToR switch Upgrade automation
- Platform for safely performing maintenances at scale

# The solution

Major components:

- Impact Analysis
- Approval Management
- Automated Execution

# Impact Analysis

- Which hosts are impacted?
- Which services are running on these hosts?
- Periodically refresh the answers to above queries

# Approval Management

- Are all services okay with the maintenance impact if it is performed **right now**?
- Automation is used to provide per-service approvals
- Periodically refresh the answer to above query

# Automated execution

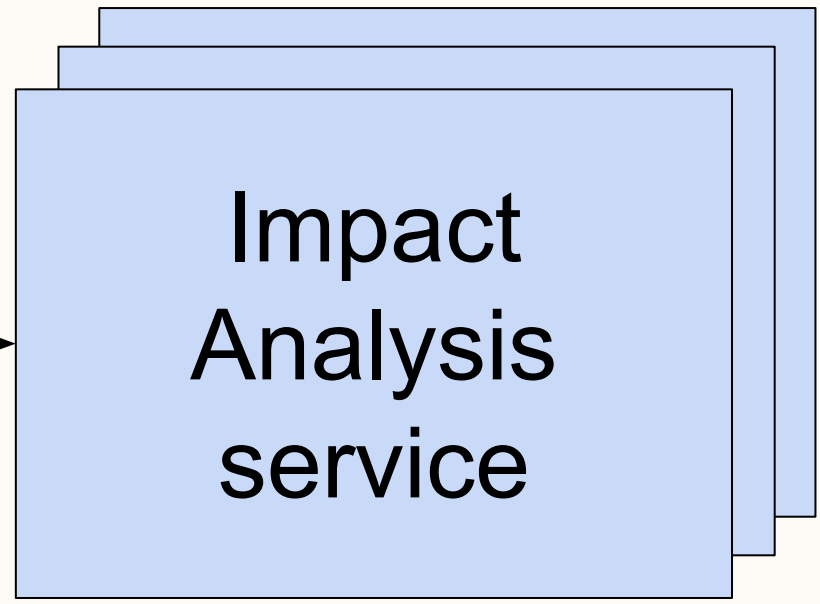
- Bring down services gracefully pre-maintenance
- Trigger the automation to perform maintenance
- Bring services back up gracefully post-maintenance





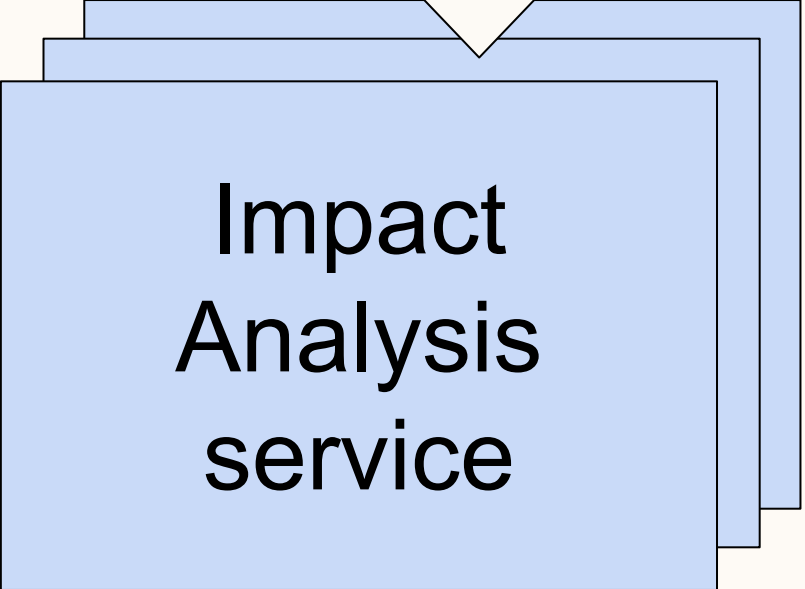
Example

foo1.switch.prod



foo1.switch.prod

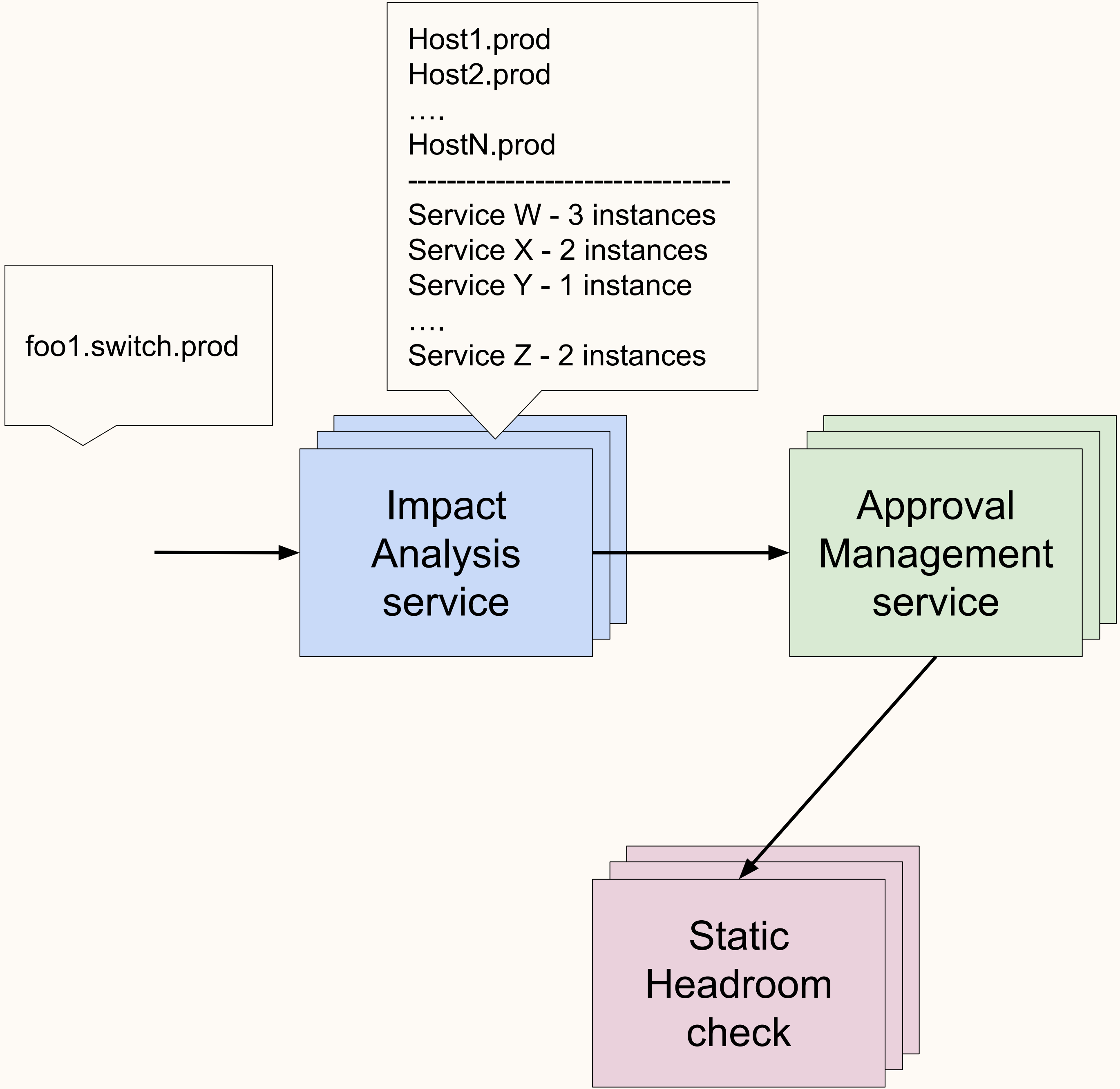
Host1.prod  
Host2.prod  
....  
HostN.prod  
-----  
Service W - 3 instances  
Service X - 2 instances  
Service Y - 1 instance  
....  
Service Z - 2 instances

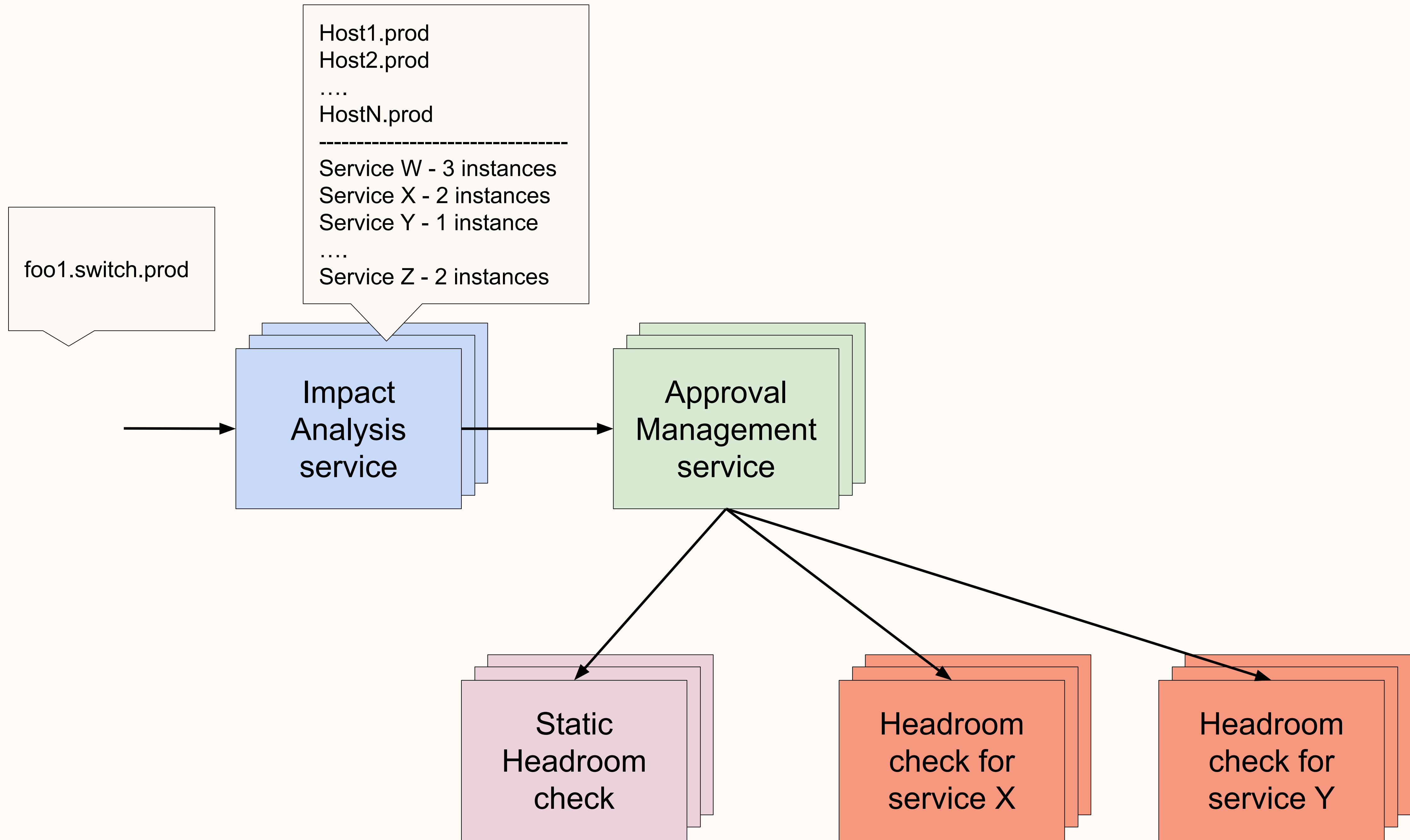


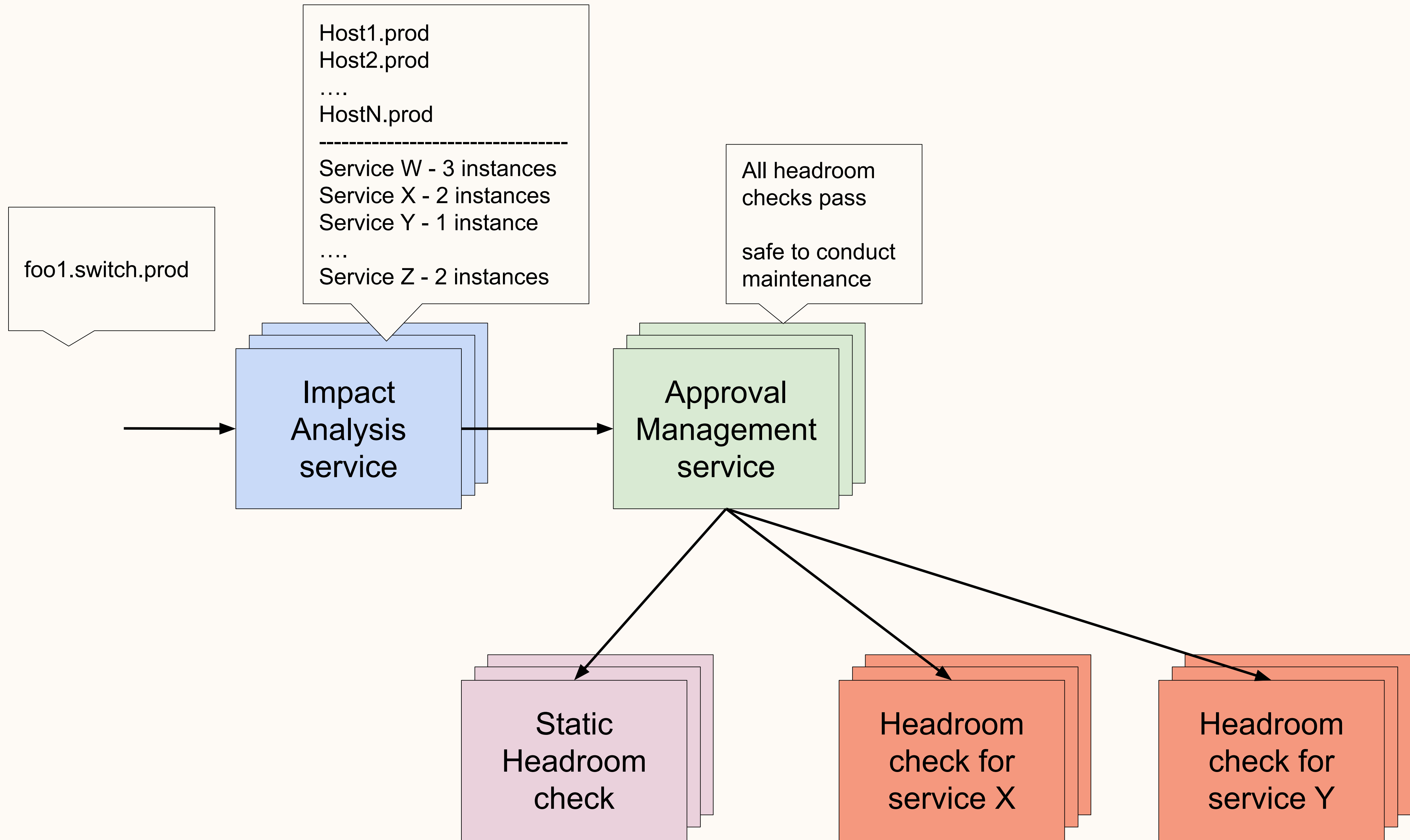
foo1.switch.prod

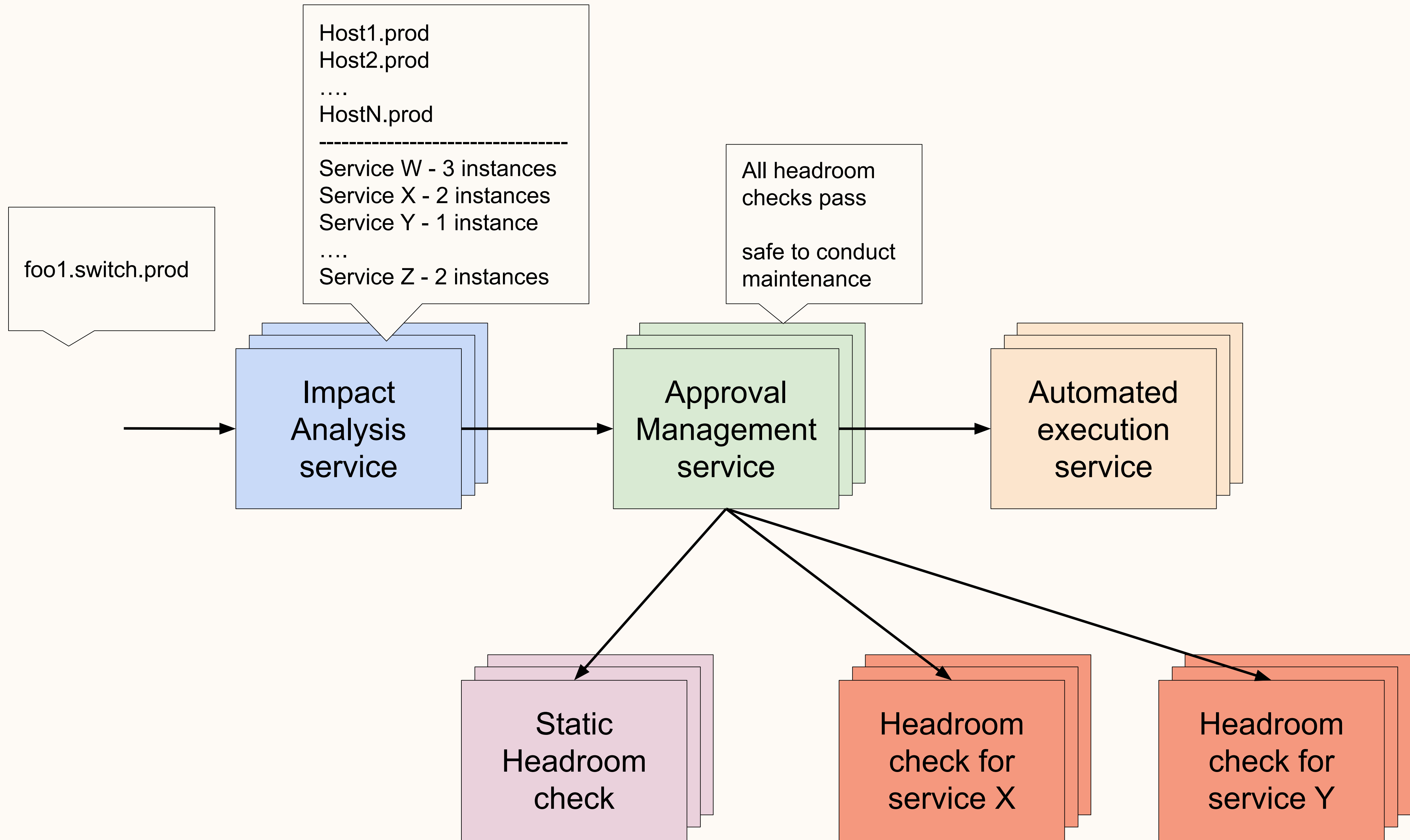
Host1.prod  
Host2.prod  
....  
HostN.prod  
-----  
Service W - 3 instances  
Service X - 2 instances  
Service Y - 1 instance  
....  
Service Z - 2 instances













# Results

- End to end maintenance for a single top of the rack switch went from taking **1 week** to less than **1 hour**
- Enabled higher order automation for fleet upgrades



What's next?

# The road ahead

- Automated OS upgrades for the whole server fleet
- Server maintenances (hardware repair/replacement)
- Quarantining servers

Thank you



<https://www.linkedin.com/in/akashvacher>