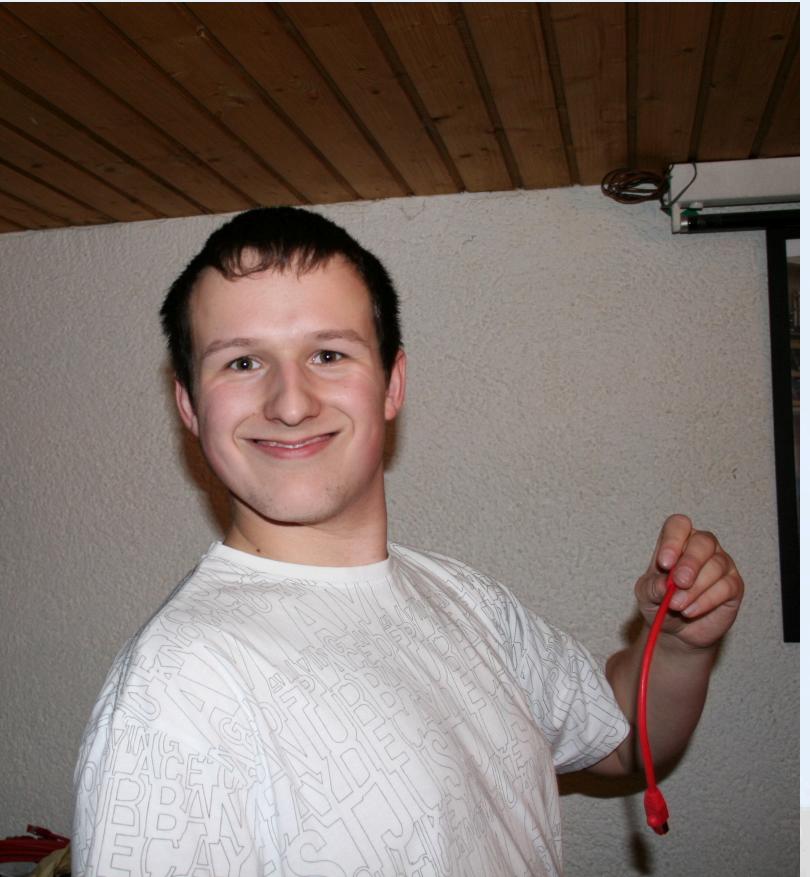


Doing mass PE upgrades in  
highly restricted environments

## \$ whoami

- Tim 'bastelfreak' Meusel
- Puppet Contributor since 2012
- Merging stuff on [Vox Pupuli](#) (Puppet Community) since 2015
- Vox Pupuli Project Management Committee member
- Senior IT Automation Consultant at [betadots](#)



Bolt

Tasks & Plans

# Bolt

## Tasks?!

- Runs tasks via ssh or WinRM on remote systems
  - CLI only



# Bolt

## Tasks?!

- Runs tasks via ssh or WinRM on remote systems
  - CLI only
- A task is a executable binary/script + json file

```
{  
  "puppet_task_version": 1,  
  "supports_noop": false,  
  "description": "Rename branch",  
  "parameters": {  
    "control_repo_branch": {  
      "description": "Control-repo branch",  
      "type": "String"  
    }  
  }  
}
```

```
#!/bin/bash  
  
pushd /tmp/control-repo || exit  
git branch -m production old_prod  
git branch -m "$PT_control_repo_branch" production
```



# Bolt

## Tasks?!

- Runs tasks via ssh or WinRM on remote systems
  - CLI only
- A task is a executable binary/script + json file
- Input and output of each task is JSON
  - makes parsing, scripting and concatenating easy



# Bolt

## Tasks?!

## Plans?!

- Bolt Plans are written in Puppet DSL (or [YAML](#))
- Plans apply puppet code, execute Bolt or Puppet functions, start tasks or other plans

```
# @param message the string we want to paste to STDOUT
plan test::foo (
  String[1] $message = 'Hi CfgMgmtCamp!',
) {
  out::message($message)
  run_task('puppet_agent::install', get_targets('all'))
}
```



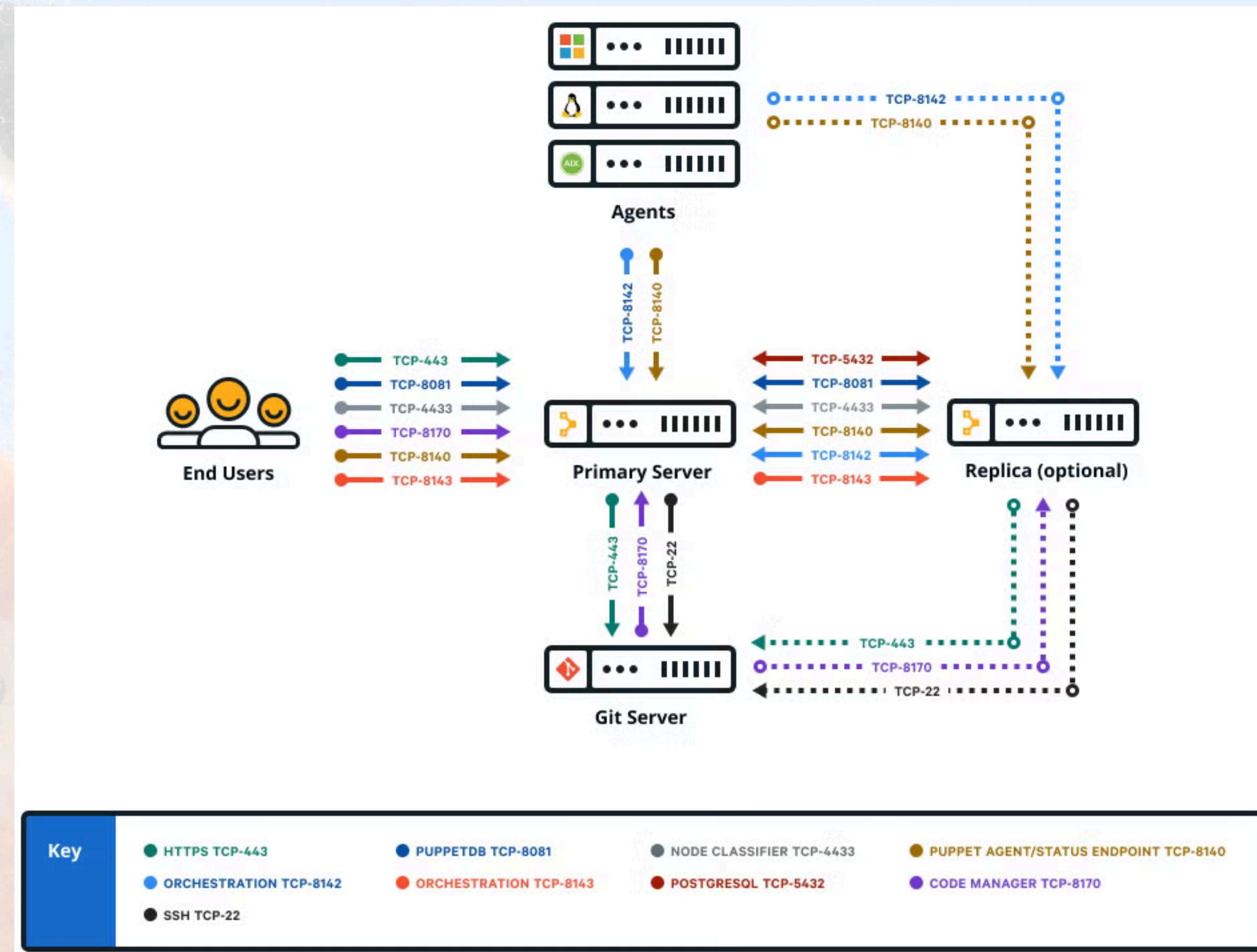
Bolt is the missing part in the ecosystem for imperative workflows and orchestration

It combines scripts and Puppet code

Puppet  
Enterprise

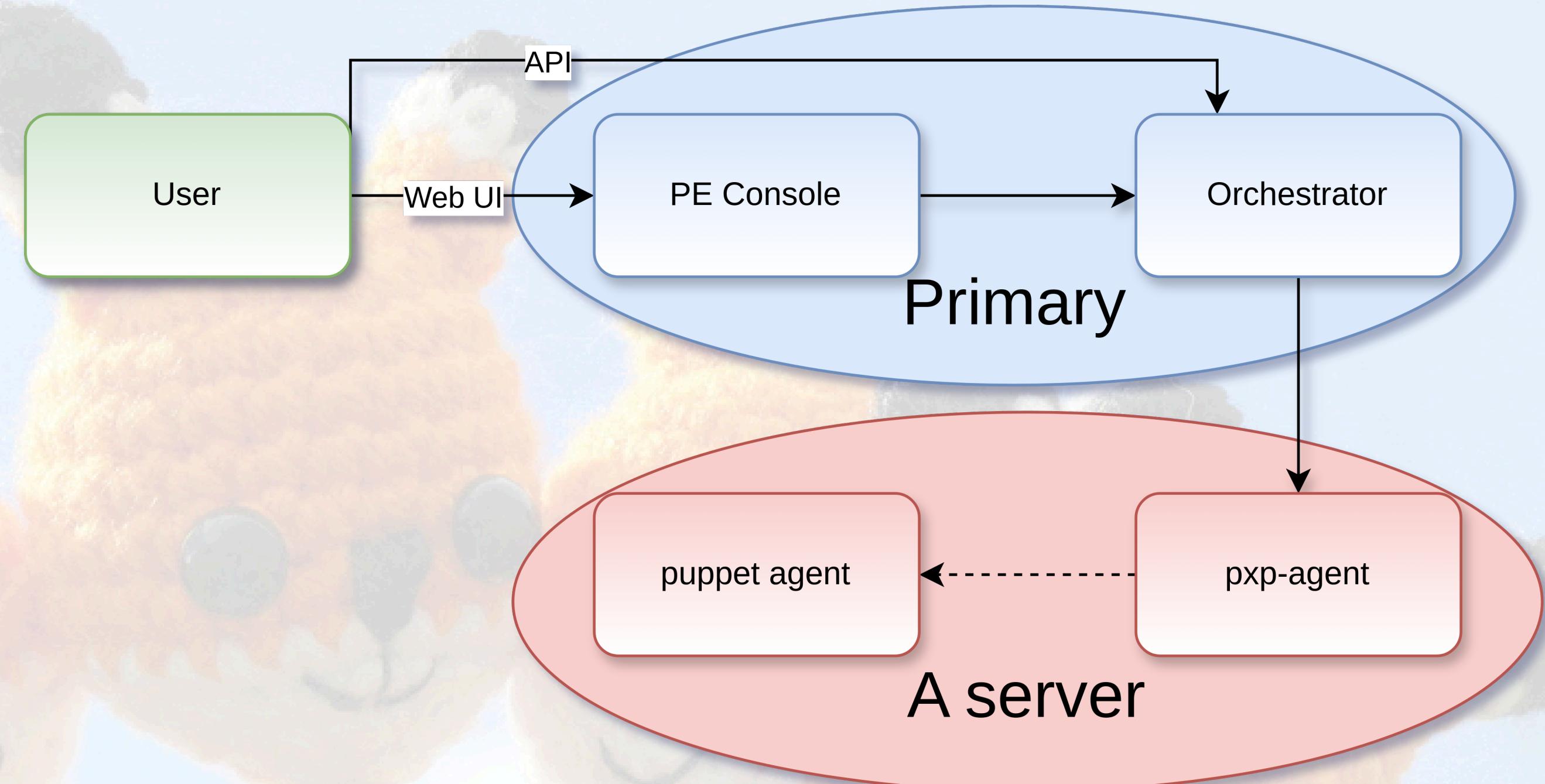
PE

What?



PE

What?



- Orchestrator can run Puppet tasks and Plans
- Basically an API & Web UI for bolt



PE

What?

Puppet Plans

- What are Puppet tasks & Plans?
- Like bolt task & Plans
- Use the PXP agent as transport



PE

What?

Puppet Plans

- What are Puppet tasks & Plans?
- Like bolt task & Plans
- Use the PXP agent as transport
- Orchestrator is closed source, PXP agent is open source
- There are subtle differences between Bolt plans and Plans in PE
  - Plans in PE have less features?! [https://www.puppet.com/docs/pe/2025.0/plans\\_limitations.html](https://www.puppet.com/docs/pe/2025.0/plans_limitations.html)



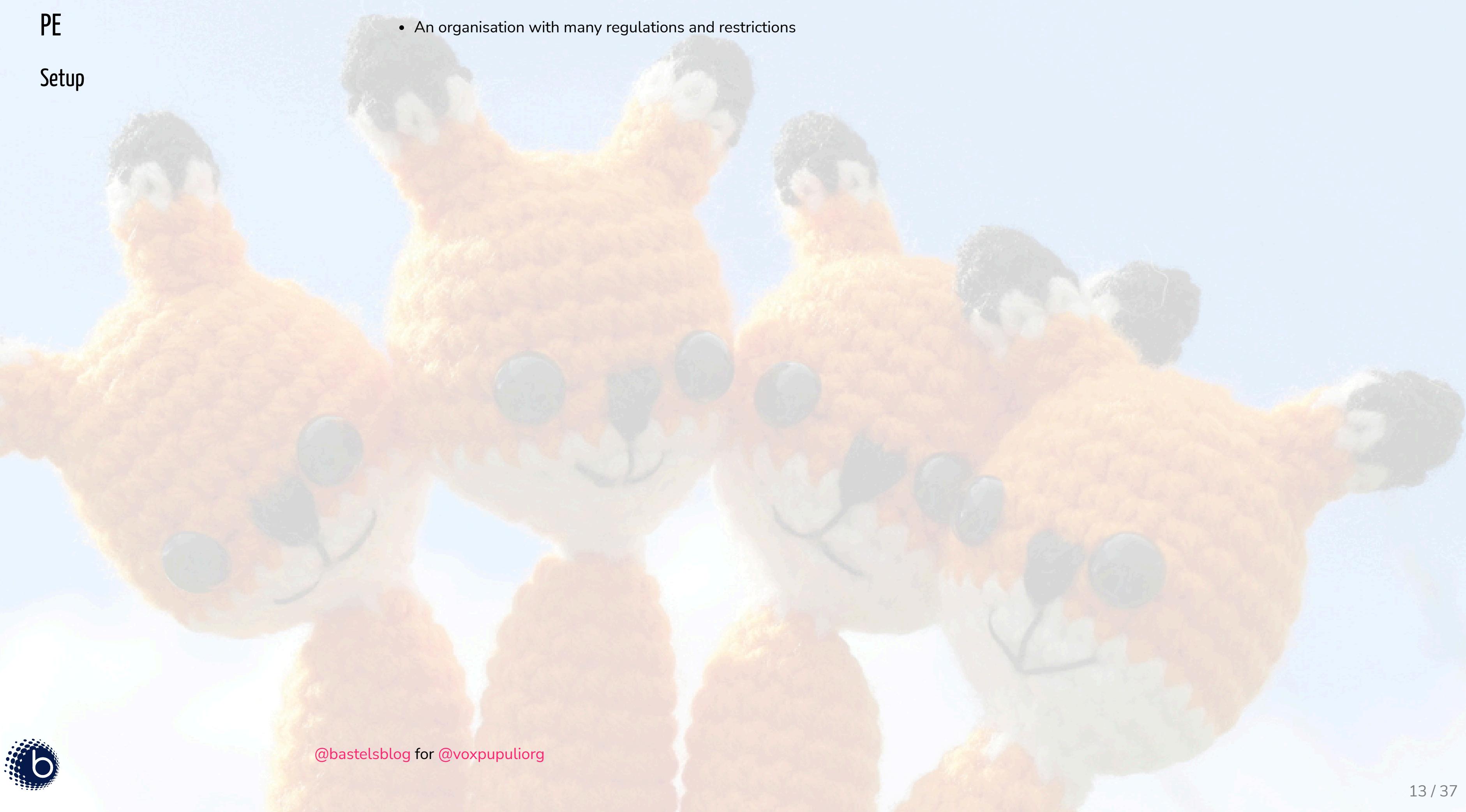
# PE in highly restricted environments

A user story

PE

- An organisation with many regulations and restrictions

## Setup



PE

## Setup

- An organisation with many regulations and restrictions
- Runs 3000 different services

Each service:

- consists of a couple of virtual machines



PE

## Setup

- An organisation with many regulations and restrictions
- Runs 3000 different services

Each service:

- consists of a couple of virtual machines
- needs to be isolated from each other



PE

## Setup

- An organisation with many regulations and restrictions
- Runs 3000 different services

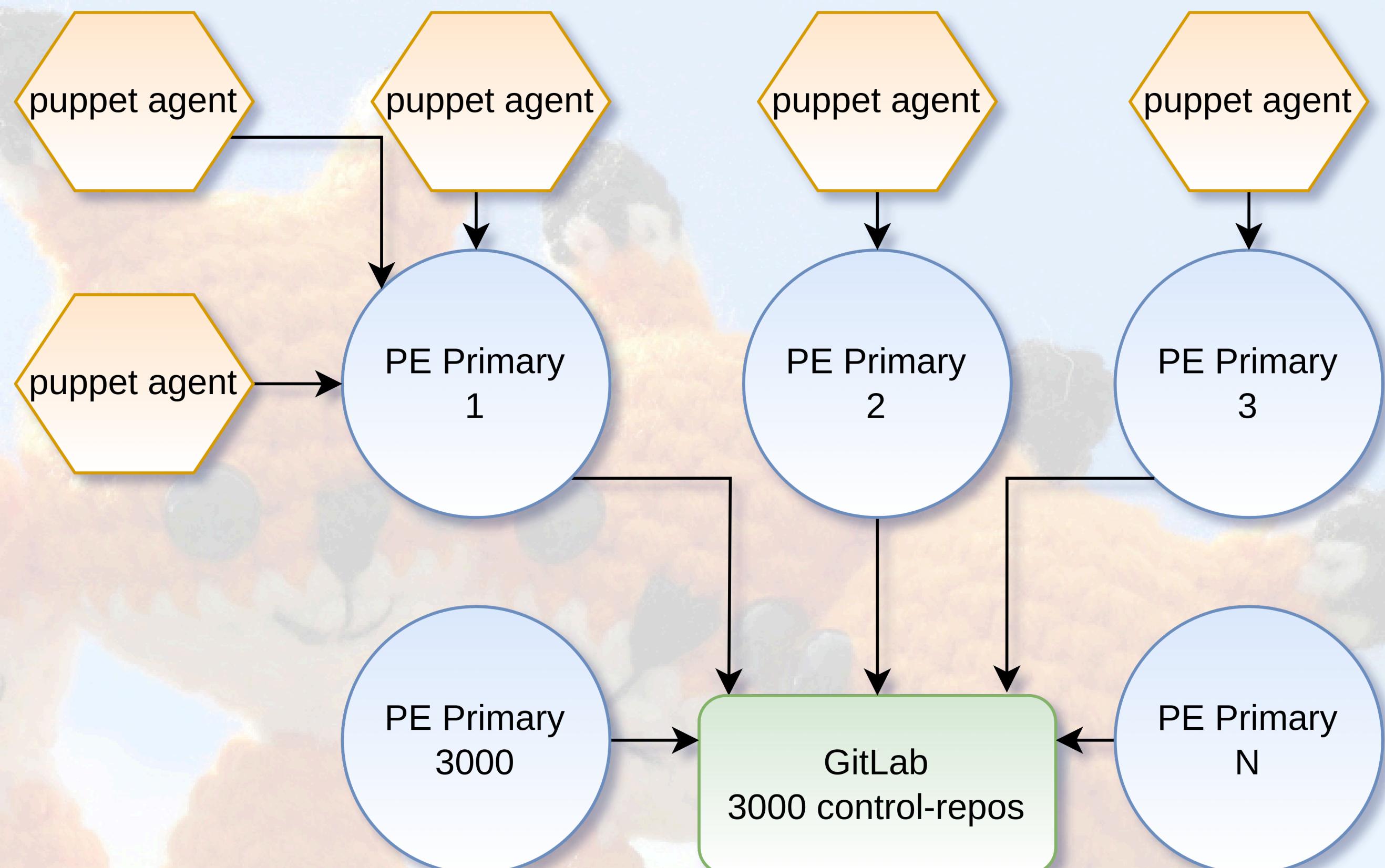
Each service:

- consists of a couple of virtual machines
- needs to be isolated from each other
- has to be managed by Puppet



PE

Setup



PE

## Setup

- 1 central GitLab
  - contains all puppet modules
  - contains 3000 control repos
- 3000 individual primaries
- 5-40 agents per primary



PE

Setup

Constraints

"security" constraints:

- No ssh access to a service



PE

Setup

Constraints

"security" constraints:

- No ssh access to a service
  - The PE APIs are available



PE

Setup

Constraints

"security" constraints:

- No ssh access to a service
  - The PE APIs are available
- No code changes for running services allowed
  - Architects wanted "immutable" services



PE

Setup

Constraints

"security" constraints:

- No ssh access to a service
  - The PE APIs are available
- No code changes for running services allowed
  - Architects wanted "immutable" services
- Of course no internet access
  - Everything needs to be mirrored internally
  - No HTTP proxy available



PE

- Update 3000 PE 2019 (Puppet 6) & PE 2021 (Puppet 7) services to PE 2023 (Puppet 8)

Setup

Constraints

Job



PE

- Update 3000 PE 2019 (Puppet 6) & PE 2021 (Puppet 7) services to PE 2023 (Puppet 8)
- Don't use ssh

Setup

Constraints

Job



PE

- Update 3000 PE 2019 (Puppet 6) & PE 2021 (Puppet 7) services to PE 2023 (Puppet 8)
- Don't use ssh
- Fully automated

Setup

Constraints

Job



PE

- Update 3000 PE 2019 (Puppet 6) & PE 2021 (Puppet 7) services to PE 2023 (Puppet 8)
- Don't use ssh
- Fully automated
- Please don't break anything

Setup

Constraints

Job



PE

- I mentioned this to Perforce employees

Setup

Constraints

Job



PE

Setup

Constraints

Job

- I mentioned this to Perforce employees
- Perforce reached out to me for a collaboration

The idea:

- I implement the upgrade somehow
- Perforce assists, helps to review PRs quickly
- Perforce can publish a whitepaper
- We publish the code as open source



PE

Setup

Constraints

Job

- I mentioned this to Perforce employees
- Perforce reached out to me for a collaboration

The idea:

- I implement the upgrade somehow
- Perforce assists, helps to review PRs quickly
- Perforce can publish a whitepaper
- We publish the code as open source
- Perforce stopped responding to all emails



Setup

Constraints

Job

Upgrade?

# Upgrading Puppet Enterprise

Upgrade your PE installation as new versions become available.

- [Upgrade PE using the installer tarball](#)

Upgrade PE infrastructure components to get the latest features and fixes. Follow the upgrade instructions for your installation type to ensure you upgrade components in the correct order. Coordinate upgrades to ensure all infrastructure nodes are upgraded in a timely manner, because agent runs and replication fail if infrastructure nodes are running a different agent version than the primary server.



Setup

Constraints

Job

Upgrade?

# Upgrading Puppet Enterprise

Upgrade your PE installation as new versions become available.

- [Upgrade PE using the installer tarball](#)

Upgrade PE infrastructure components to get the latest features and fixes. Follow the upgrade instructions for your installation type to ensure you upgrade components in the correct order. Coordinate upgrades to ensure all infrastructure nodes are upgraded in a timely manner, because agent runs and replication fail if infrastructure nodes are running a different agent version than the primary server.

- TL;DR: "ssh to the Primary and download a tarball"



PE

What's in the tarball?

- Free to download for everybody

Setup

Constraints

Job

Upgrade?



PE

Setup

Constraints

Job

Upgrade?

What's in the tarball?

- Free to download for everybody
- for example `puppet-enterprise-2021.7.8-el-8-x86_64.tar.gz`
  - PE version specific
  - OS specific



PE

Setup

Constraints

Job

Upgrade?

What's in the tarball?

- Free to download for everybody
- for example `puppet-enterprise-2021.7.8-el-8-x86_64.tar.gz`
  - PE version specific
  - OS specific
- contains a yum repo with some rpms and a long bash script



PE

Setup

Constraints

Job

Upgrade?

What's in the tarball?

- Free to download for everybody
- for example `puppet-enterprise-2021.7.8-el-8-x86_64.tar.gz`
  - PE version specific
  - OS specific
- contains a yum repo with some rpms and a long bash script
- one package contains puppet modules



PE

Setup

Constraints

Job

Upgrade?

What's in the tarball?

- Free to download for everybody
- for example `puppet-enterprise-2021.7.8-el-8-x86_64.tar.gz`
  - PE version specific
  - OS specific
- contains a yum repo with some rpms and a long bash script
- one package contains puppet modules
- bash script install the rpms & runs puppet apply



PE

- License prohibits putting the puppet modules into my GitLab
- License prohibits putting the rpms on my local mirror

Setup

Constraints

Job

Upgrade?



PE

Setup

Constraints

Job

Upgrade?

- License prohibits putting the puppet modules into my GitLab
- License prohibits putting the rpms on my local mirror
- With the rpms, the upgrade would be so much easier to automate



# Upgrading Puppet Enterprise

Upgrade your PE installation as new versions become available.

- [\*\*Upgrade PE using the installer tarball\*\*](#)

Upgrade PE infrastructure components to get the latest features and fixes. Follow the upgrade instructions for your installation type to ensure you upgrade components in the correct order. Coordinate upgrades to ensure all infrastructure nodes are upgraded in a timely manner, because agent runs and replication fail if infrastructure nodes are running a different agent version than the primary server.

- [\*\*Upgrade PE using PIM\*\*](#)

Puppet Installation Manager (PIM) supports the upgrading of Puppet Enterprise (PE) for all supported installation architectures. For an interactive experience, choose the guided upgrade process and follow the steps in your terminal. Alternatively, if you do not require guidance, you can run your upgrade from the PIM command line by passing a JSON file containing your installation parameters.

- [https://www.puppet.com/docs/pe/latest/upgrading\\_pe.html](https://www.puppet.com/docs/pe/latest/upgrading_pe.html)
- PIM is a frontend around Bolt & [puppetlabs-peadm](#)



PE

- PIM is a frontend around Bolt & [puppetlabs-peadm](#)
- PEADM: A module with tasks and plans to install/modify/upgrade PE

Setup

Constraints

Job

Upgrade?



PE

Setup

Constraints

Job

Upgrade?

- PIM is a frontend around Bolt & [puppetlabs-peadm](#)
- PEADM: A module with tasks and plans to install/modify/upgrade PE
- PEADM wraps the original installer tarball



PE

Setup

Constraints

Job

Upgrade?

- PIM is a frontend around Bolt & [puppetlabs-peadm](#)
- PEADM: A module with tasks and plans to install/modify/upgrade PE
- PEADM wraps the original installer tarball
- PEADM is compatible with PE 2019.8.1 and newer



PE

Setup

Constraints

Job

Upgrade?

Upgrade!

# PE installer

- We cannot run the installer directly, because we don't have ssh access



PE

- The Orchestrator has an API to start plans
- Can it upgrade itself?

Setup

Constraints

Job

Upgrade?

Upgrade!



PE

Setup

Constraints

Job

Upgrade?

Upgrade!

- The Orchestrator has an API to start plans
- Can it upgrade itself?
  - No



PE

Setup

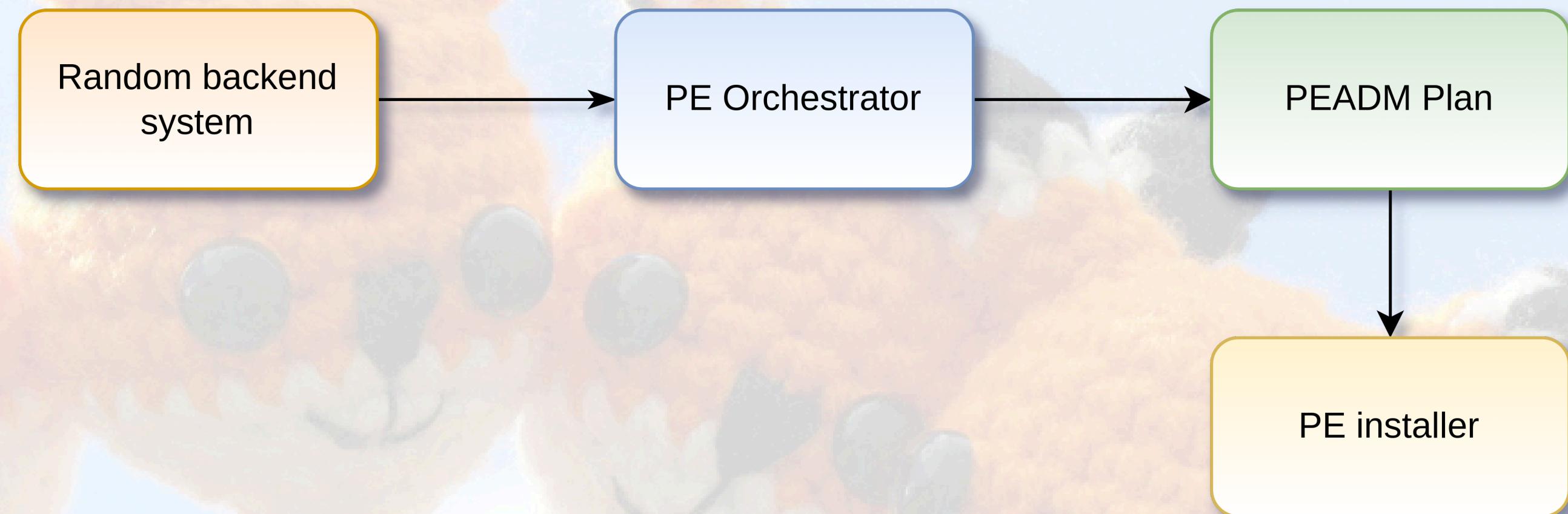
Constraints

Job

Upgrade?

Upgrade!

- The Orchestrator has an API to start plans
- Can it upgrade itself?
  - No
  - orchestrator rpm needs to be upgraded and service needs to be restarted
  - This will deadlock or abort the PEADM plan



PE

- The module `puppetlabs/service` has tasks to start/stop/inspect a service

Setup

Constraints

Job

Upgrade?

Upgrade!



PE

Setup

Constraints

Job

Upgrade?

Upgrade!

- The module `puppetlabs/service` has tasks to start/stop/inspect a service
- We could write a systemd unit that starts bolt with `Type=exec`



PE

Setup

Constraints

Job

Upgrade?

Upgrade!

- The module `puppetlabs/service` has tasks to start/stop/inspect a service
- We could write a systemd unit that starts bolt with `Type=exec`
  - Then systemd will run bolt in the background (asynchronously)



PE

Setup

Constraints

Job

Upgrade?

Upgrade!

- The module `puppetlabs/service` has tasks to start/stop/inspect a service
- We could write a systemd unit that starts bolt with `Type=exec`
  - Then systemd will run bolt in the background (asynchronously)

```
# /etc/systemd/system/pedadmmig@.service
# THIS FILE IS MANAGED BY PUPPET
[Unit]
Description=run bolt plans in project pedadmmig

[Service]
Type=exec
ExecStart=/opt/puppetlabs/bin/bolt plan run %i --params @/opt/pedadmmig/%i.json
User=pedadmmig
Group=pedadmmig
WorkingDirectory=/opt/pedadmmig
# don't add RemainAfterExit,
# then we cannot track the state via puppet anymore after bolt started
```



PE

Setup

Constraints

Job

Upgrade?

Upgrade!

- A template service unit contains a single @ and accepts strings afterwards
- This allows us to pass plan names to our unit
- `systemctl {status,start} peadmmig@profiles::convert.service`



PE

Setup

Constraints

Job

Upgrade?

Upgrade!

- A template service unit contains a single @ and accepts strings afterwards
- This allows us to pass plan names to our unit
- `systemctl {status,start} peadmmig@profiles::convert.service`
- PE doesn't have bolt installed by default
- In May 2024 we asked if PE could provide bolt by default
  - Still no response. It was promised multiple times that someone will take a look at the issue



PE

Setup

Constraints

Job

Upgrade?

Upgrade!

- A **template service unit** contains a single @ and accepts strings afterwards
- This allows us to pass plan names to our unit
- `systemctl {status,start} peadmmig@profiles::convert.service`
- PE doesn't have bolt installed by default
  - In **May 2024** we asked if PE could provide bolt by default
    - Still no response. It was promised multiple times that someone will take a look at the issue
- Vox Pupuli now has a module to install bolt: [forge.puppet/com/puppet/bolt](#)
  - The module can create the systemd template service unit + all required configuration files



PE

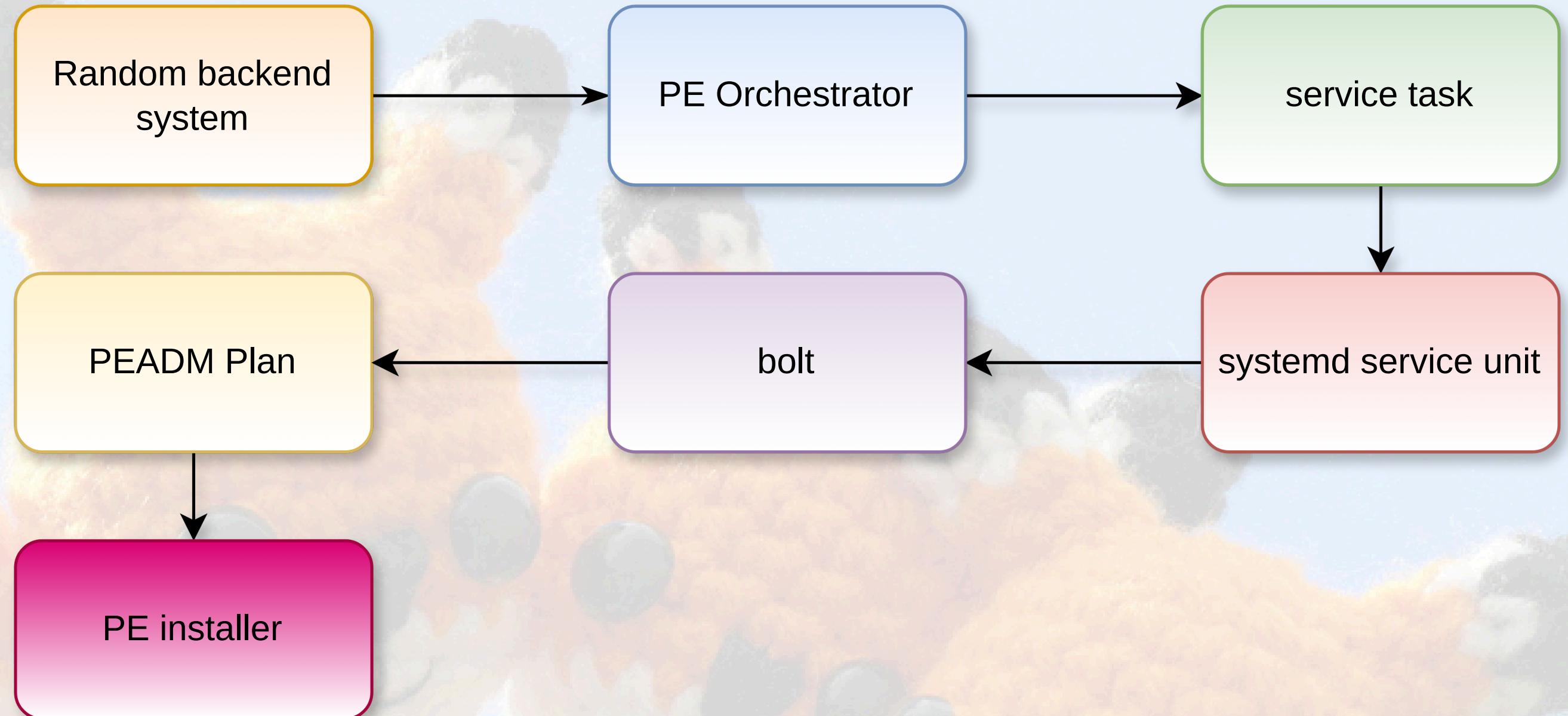
Setup

Constraints

Job

Upgrade?

Upgrade!



PE

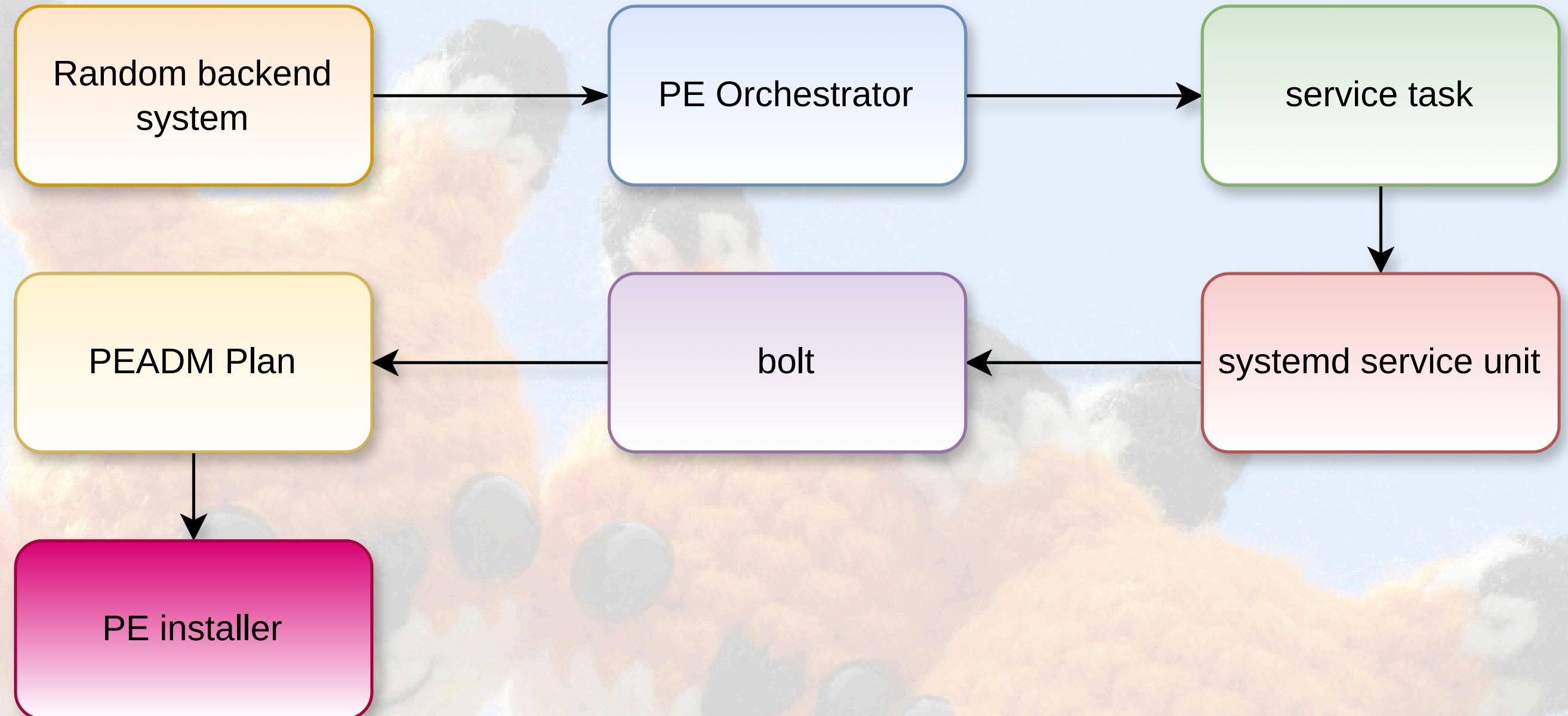
Setup

Constraints

Job

Upgrade?

Upgrade!



- During development, this worked surprisingly well



PE

- We need our own plan that does some sanity/health checks
  - If everything is fine, our plan starts the `peadm::upgrade` plan

Setup

Constraints

Job

Upgrade?

Upgrade!

Deployment?



PE

Setup

Constraints

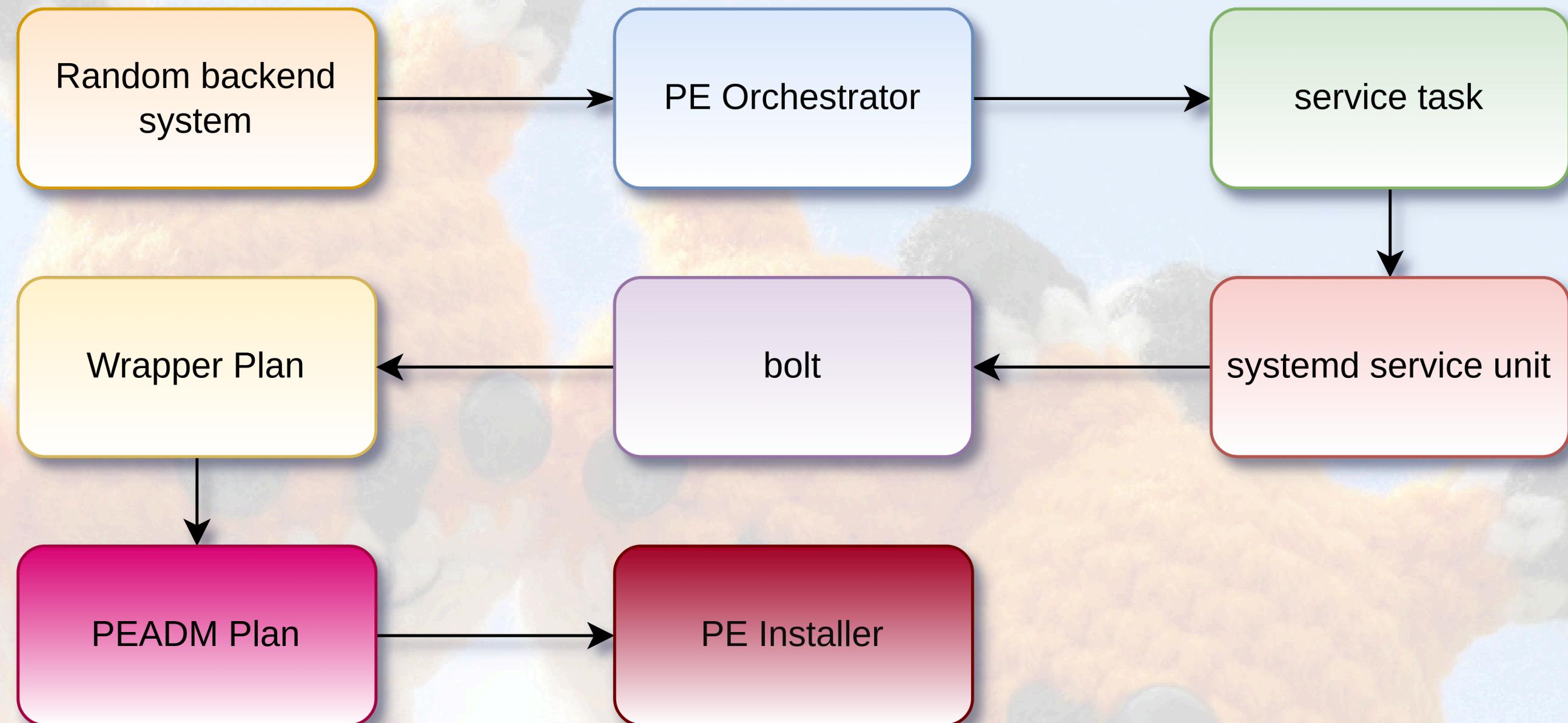
Job

Upgrade?

Upgrade!

Deployment?

- We need our own plan that does some sanity/health checks
  - If everything is fine, our plan starts the peadm::upgrade plan



PE

Setup

Constraints

Job

Upgrade?

Upgrade!

Deployment?

- We need to run `peadm::convert` before `peadm::upgrade`
  - This will replace the TLS certificate on your primary



PE

Setup

Constraints

Job

Upgrade?

Upgrade!

Deployment?

- We need to run `peadm::convert` before `peadm::upgrade`
  - This will replace the TLS certificate on your primary
- We need to deploy our own plans, but code deployments aren't allowed



PE

Setup

Constraints

Job

Upgrade?

Upgrade!

Deployment?

- We need to run `peadm::convert` before `peadm::upgrade`
  - This will replace the TLS certificate on your primary
- We need to deploy our own plans, but code deployments aren't allowed
- "No code deployments that might impact normal puppet agent operations"
- PE supports multiple control repositories
  - We can add another control repo, that only contains tasks/plan/modules for PE upgrades
  - No agent will make a puppet run in this environment



PE

Setup

Constraints

Job

Upgrade?

Upgrade!

Deployment?

Preparation

- PE has two ways to specify a control repo:
- A single control repo

```
puppet_enterprise::profile::master::r10k_remote: 'https://github.com/testcontrolrepo.git'
```

- N control repos

```
puppet_enterprise::master::code_manager::sources:  
  foo:  
    remote: 'https://github.com/bastelfreak/testcontrolrepo.git'  
    prefix: false  
  baz:  
    remote: 'https://github.com/voxpupuli/controlrepo'  
    prefix: true
```



PE

- You should not specify both Hiera Keys

Setup

Constraints

Job

Upgrade?

Upgrade!

Deployment?

Preparation



PE

Setup

Constraints

Job

Upgrade?

Upgrade!

Deployment?

Preparation

- You should not specify both Hiera Keys
- PE Installer takes a config file (`pe.conf`) that accepts Hiera keys



PE

Setup

Constraints

Job

Upgrade?

Upgrade!

Deployment?

Preparation

- You should not specify both Hiera Keys
- PE Installer takes a config file (`pe.conf`) that accepts Hiera keys
- PE Console can also serve Hiera data



PE

Setup

Constraints

Job

Upgrade?

Upgrade!

Deployment?

Preparation

- You should not specify both Hiera Keys
- PE Installer takes a config file (`pe.conf`) that accepts Hiera keys
- PE Console can also serve Hiera data
- A cronjob dumps PE Console data to disk (`user_data.conf`)



PE

Setup

Constraints

Job

Upgrade?

Upgrade!

Deployment?

Preparation

- You should not specify both Hiera Keys
- PE Installer takes a config file (`pe.conf`) that accepts Hiera keys
- PE Console can also serve Hiera data
- A cronjob dumps PE Console data to disk (`user_data.conf`)
- `pe.conf` and `user_data.conf` use HOCON format



PE

Setup

Constraints

Job

Upgrade?

Upgrade!

Deployment?

Preparation

- You should not specify both Hiera Keys
- PE Installer takes a config file ([pe.conf](#)) that accepts Hiera keys
- PE Console can also serve Hiera data
  - A cronjob dumps PE Console data to disk (user\_data.conf)
  - pe.conf and user\_data.conf use HOCON format
  - Puppet core/stdlib have no function to write HOCON



PE

Setup

Constraints

Job

Upgrade?

Upgrade!

Deployment?

Preparation

- Every deployed service has their own control repo branch
  - No agents runs in a production environment



PE

Setup

Constraints

Job

Upgrade?

Upgrade!

Deployment?

Preparation

- Every deployed service has their own control repo branch
  - No agents runs in a production environment
- PE Console in some versions requires a production environment to interact with plans



PE

Setup

Constraints

Job

Upgrade?

Upgrade!

Deployment?

Preparation

- Every deployed service has their own control repo branch
  - No agents runs in a production environment
- PE Console in some versions requires a production environment to interact with plans
- PEADM assumes that a production branch exists
  - This isn't listed as a requirement anywhere



PE

Setup

Constraints

Job

Upgrade?

Upgrade!

Deployment?

Preparation

- Every deployed service has their own control repo branch
  - No agents runs in a production environment
- PE Console in some versions requires a production environment to interact with plans
- PEADM assumes that a production branch exists
  - This isn't listed as a requirement anywhere
  - We reported this in July 2024
  - This was a blocker for the biggest PE customer I ever met



PE

Setup

Constraints

Job

Upgrade?

Upgrade!

Deployment?

Preparation

- Every deployed service has their own control repo branch
  - No agents runs in a production environment
- PE Console in some versions requires a production environment to interact with plans
- PEADM assumes that a production branch exists
  - This isn't listed as a requirement anywhere
  - We reported this in July 2024
  - This was a blocker for the biggest PE customer I ever met
  - We did our own debugging and proposed a fix in August 2024



PE

Setup

Constraints

Job

Upgrade?

Upgrade!

Deployment?

Preparation

- Every deployed service has their own control repo branch
  - No agents runs in a production environment
- PE Console in some versions requires a production environment to interact with plans
- PEADM assumes that a production branch exists
  - This isn't listed as a requirement anywhere
  - We reported this in July 2024
  - This was a blocker for the biggest PE customer I ever met
  - We did our own debugging and proposed a fix in August 2024
  - This is still not released, Engineering and product owner ignore all slack/email messages
  - We had escalation meetings with our sales rep and some engineers in September



PE

Setup

Constraints

Job

Upgrade?

Upgrade!

Deployment?

Preparation

- Required Hiera keys when the Primary isn't running in production

```
pe_install::install::classification::pe_node_group_environment: 'foo'  
puppet_enterprise::master::recover_configuration::pe_environment: 'foo'
```

- Only required for inplace updates
- [puppet.com/docs/pe/latest/upgrading\\_pe.html](https://puppet.com/docs/pe/latest/upgrading_pe.html)



PE

Setup

Constraints

Job

Upgrade?

Upgrade!

Deployment?

Preparation

- Puppet agent sometimes starts a run in production environment
- ENC tells the agent the correct environment
- puppet agent starts again



PE

Setup

Constraints

Job

Upgrade?

Upgrade!

Deployment?

Preparation

- Puppet agent sometimes starts a run in production environment
- ENC tells the agent the correct environment
- puppet agent starts again

```
ini_setting { 'puppet.conf environment':  
    ensure  => 'present',  
    path    => '/etc/puppetlabs/puppet/puppet.conf',  
    section => 'agent',  
    setting => 'environment',  
    value   => $env,  
}
```



PE

Setup

Constraints

Job

Upgrade?

Upgrade!

Deployment?

Preparation

- Puppet agent sometimes starts a run in production environment
- ENC tells the agent the correct environment
- puppet agent starts again

```
ini_setting { 'puppet.conf environment':  
    ensure  => 'present',  
    path    => '/etc/puppetlabs/puppet/puppet.conf',  
    section => 'agent',  
    setting => 'environment',  
    value   => $env,  
}
```

For properly managing Puppet:

- [forge.puppet.com/puppetlabs/puppet\\_agent](https://forge.puppet.com/puppetlabs/puppet_agent)
  - By default ignores the puppet.conf on your PE Infra
- [forge.puppet.com/theforeman/puppet](https://forge.puppet.com/theforeman/puppet)



- how to deploy code
- different preparation steps
- cleanup node groups / pe.conf
- ensure PEADM uses correct node group
- the issue with the installer URL
- stderr redirection <https://github.com/puppetlabs/puppetlabs-peadm/pull/523>
- duplicated tasks for running puppet agent
- duplicated fact detection <https://github.com/puppetlabs/puppetlabs-peadm/pull/459>
- Puppet is very flexible and suitable for different usecases
- There are many hidden gems and extensions available
- More docs from Puppet/Perforce for edgecases & advanced users would be awesome!
- For feedback: `bastelfreak` on [slack.puppet.com/Libera.Chat](https://slack.puppet.com/Libera.Chat) IRC or [tim@bastelfreak.de](mailto:tim@bastelfreak.de)
- This talk and previous ones: [github.com/bastelfreak/talks](https://github.com/bastelfreak/talks)

Thanks for your attention!

]



@bastelsblog for @voxpupuliorg