

# PDP-5: Organization of PNDA CLI & creational repositories

## Motivation

The [PNDA CLI](#) initially only supported creating PNDA on AWS using the Cloud Formation API. Subsequently, support was added for skipping the Cloud Formation parts and installing PNDA on an existing set of servers (the 'existing machines' approach).

An entirely different code base is used to install PNDA on Openstack using HEAT (this is located in the [pnda-heat-templates](#) repository).

There are maintainability benefits to having a single code base for all CLIs - for example a 100% consistent CLI syntax, shared code to validate user input and so on.

So we would like to move the current PNDA CLI towards a unified CLI that contains specific implementations to support each target deployment environment - AWS, HEAT, Existing machines, and any that may come along in future, while sharing other logic such as user input, configuration and validation. The first step towards this is to refactor the PNDA CLI into a more structured architecture, with the AWS and Existing machines code separated out into their own modules.

## Proposal

- Refactor pnda-cli into clear front ends and back ends
  - Front ends handle command line interface or REST API, for example
  - Back ends handle creating PNDA on different platforms
- Factor out configuration validation and user input validation cleanly
  - Revise CLI to follow normal conventions
- Migrate pnda-heat-templates to be a pnda-cli back end and deprecate this repository
- Longer term, review e.g. Terraform as a cross-cloud approach to handling back end functionality

## Plan

Phase 1 (complete)

- Separate out user input validation
- Create two classes CloudFormationBackend and ExistingMachinesBackend that inherit from a BaseBackend class.
- Move all functionality relating to deployment under BaseBackend, with overrides from CloudFormationBackend and ExistingMachinesBackend where target specific functionality is required.

Phase 2

- Refactor pnda-env.yaml into below, to get little more organized for supporting multiple cloud infrastructures
  - pnda-cloud-infra.yaml – introduce 2 categories of config parameters (AWS & Openstack) and add respective network, connectivity & image details under them
  - pnda-env.yaml – to have common config parameters across cloud infrastructures with appropriate categories
- Migrate functionality in pnda-heat-templates to pnda-cli and deprecate the former repository
  - Create class HeatBackend that inherit from a BaseBackend class
  - Align instance numbers with AWS by removing a dedicated vm for salt-master in pico setup
  - Align instance Key usage with AWS by having only one key for bastion & other nodes
  - Align kafka ingest interface config similar to AWS
  - No impact on pnda-dib-elements repository

Phase 3

- TBD