Easy Installation using Terraform and Ansible

Spectrum Scale German User Meeting 2022
Cologne, Germany – October 19th, 2022

Achim Christ (IBM)
Muthu Muthiah (IBM)

Disclaimer

IBM's statements regarding its plans, directions, and intent are subject to change or withdrawal without notice at IBM's sole discretion. Information regarding potential future products is intended to outline our general product direction and it should not be relied on in making a purchasing decision. The information mentioned regarding potential future products is not a commitment, promise, or legal obligation to deliver any material, code, or functionality. The development, release, and timing of any future features or functionality described for our products remains at our sole discretion.

IBM reserves the right to change product specifications and offerings at any time without notice. This publication could include technical inaccuracies or typographical errors. References herein to IBM products and services do not imply that IBM intends to make them available in all countries.

Motivation & Goals

- Make it easy to install, configure & update Spectrum Scale environments, by enabling an "*Infrastructure as Code*" (IaC) management model.

- Provide reusable code artifacts for industry standard, open source tools:
  - Cloud agnostic infrastructure provisioning through HashiCorp Terraform
  - Spectrum Scale installation and configuration through Red Hat Ansible

→ Consistent tooling independent of the underlying infrastructure

What's Infrastructure as Code?

- Apply software (development) practices to infrastructure management:
  - ☑ Clearly defined state – '[Single Source of Truth]'
  - ☑ Combine with version control – '[GitOps]'
  - ☑ Ability to version / roll-back changes
  - ☑ Keep test / staging / production in sync
  - ☑ Efficiently scale-up / scale-down components

  …

***Infrastructure as code** is the process of managing and provisioning computer data centers through machine-readable definition files, rather than physical hardware configuration or interactive configuration tools.*
[Wikipedia]

IaC Tooling

- Open source, backed with commercial offerings
- CLI (+ optional GUI)
- Configuration management,
  automation of operational tasks
- Declarative definition language
  - Current state ↔ desired state
- Active ecosystem, large user base,
  large amounts of community content
- Ansible vs. Terraform [link]

HashiCorp Terraform [link]

- *“Automate Infrastructure on Any Cloud”*
- HashiCorp Configuration Language (HCL)
- Provision ‘resources’ (using ‘providers’)
- Terraform Registry: https://registry.terraform.io/
- Sweet spot:
  provision (cloud) infrastructure resources

Red Hat Ansible [link]

- *“Simple IT Automation”*
- Yet Another Markup Language (YAML)
- Execute idempotent ‘tasks’ (using ‘modules’)
- Ansible Galaxy: https://galaxy.ansible.com/
- Sweet spot:
  manage service configuration at scale

Spectrum Scale Automated Deployment “Before”

SDI / ECE

Scale install scripts

Scale Install Toolkit
- Core Install & deploy
- Protocol Install & deploy
- Upgrade Core & Protocols
- ECE/FAL/Call home/Perf. Mon.

Implementation:
Python, Ruby Chef DSL

Cross cluster Orchestration:
Chef

ESS

Hardware setup scripts

Scale install scripts

Scale Install Toolkit
- Protocol install

Implementation:
Python, Shell scripts

Cross cluster Orchestration:
xCAT

Cloud

Cloud provisioning scripts

Scale install scripts

Scale Install Toolkit
- Upgrade

Implementation:
Shell Scripts

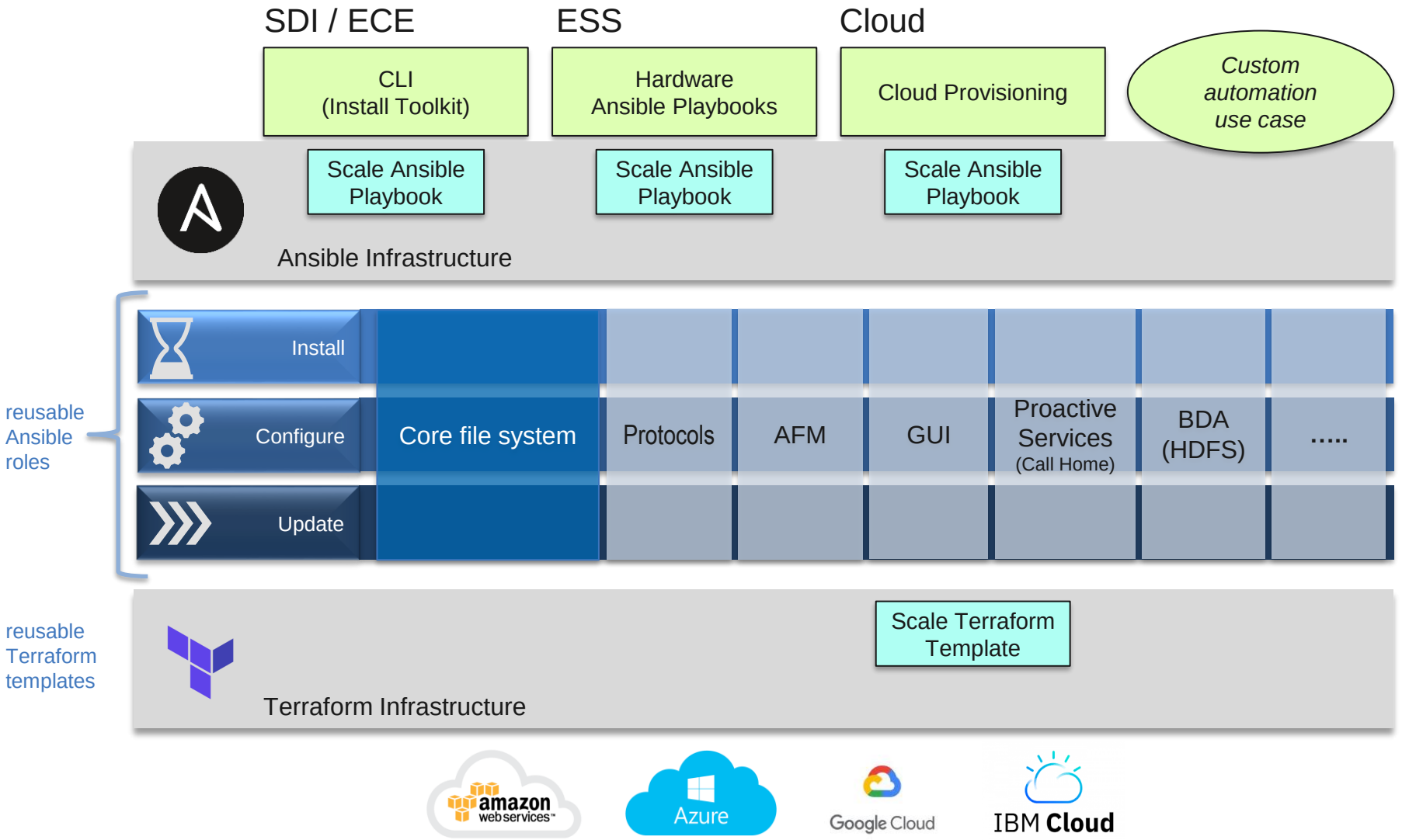
Cross cluster Orchestration
Amazon Cloud Formation

Dev & Test

Scale Install Toolkit
- Core Install & deploy
- Protocol Install & deploy

Implementation:
Shell Script, Python, Chef DSL, Open Stack

Spectrum Scale Automated Deployment “After”



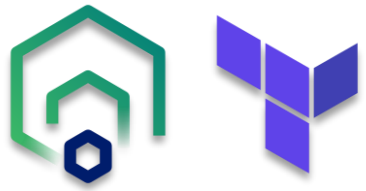
Spectrum Scale Integration Points

Spectrum Scale ↔ Ansible integration:
https://github.com/ibm/ibm-spectrum-scale-install-infra
(Apache 2.0 license)

Fine-grained feature selection (modular roles)

Scope:

- Initial deployment / bootstrap
- Lifecycle management
  - Add & remove nodes
  - Add & remove disks
  - Updates



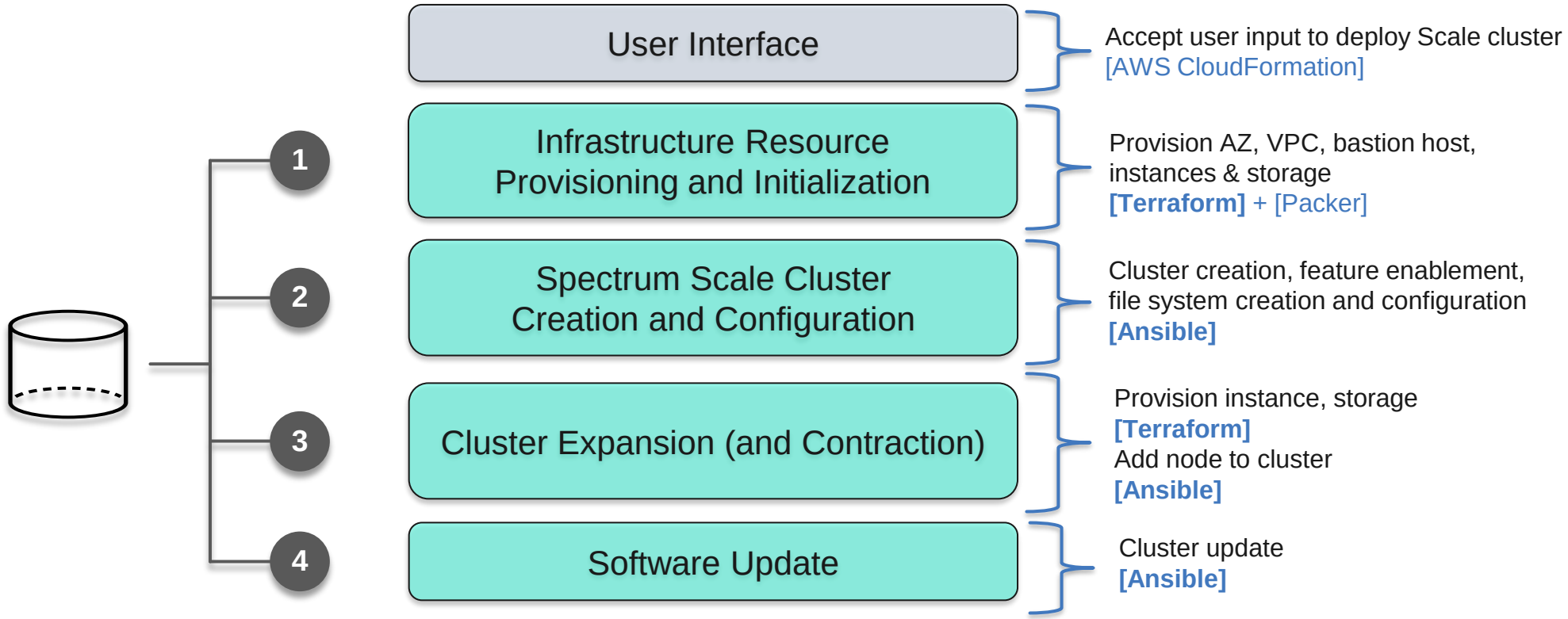
Spectrum Scale ↔ Terraform integration:
https://github.com/ibm/ibm-spectrum-scale-cloud-install
(Apache 2.0 license)

Cloud-agnostic: AWS, IBM Cloud, (GCP, Azure)

Scope:

- Preset deployment patterns:
  - Single / multiple availability zones
  - Single / multiple clusters
  - Re-use existing VPC
- Provision: (VPC), bastion host, private network, EC2 instances, EBS volumes
- Optionally run Ansible roles for cluster creation & filesystem configuration

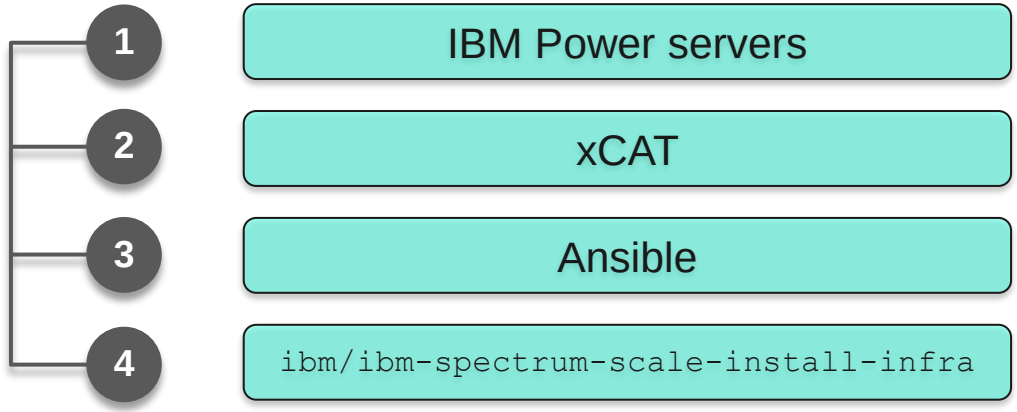
Putting it all together (1 of 3)



IBM Spectrum Scale @ AWS Marketplace
https://aws.amazon.com/marketplace/pp/prodview-sxlolie3dpncg

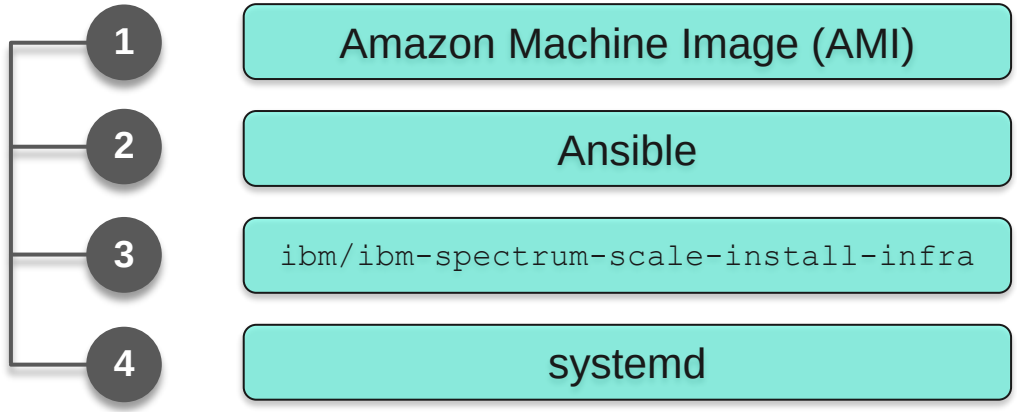
Putting it all together (2 of 3)

- University & research customer @ Switzerland
- Goal
  - Efficiently deploy & manage large protocol clusters consistently
  - Refresh IBM Power server hardware
- Challenges
  - 12 protocol nodes (SMB & NFS) with AD integration
  - Complex network / firewall / hardening rules
  - Historically grown: slightly different configuration on each node
  - Migrate from Big Endian → Little Endian
- Solution
  - IBM Power 9 servers
  - Initialization & OS installation through xCAT
  - OS configuration (security, compliance) through Ansible
  - Spectrum Scale installation through Ansible
  - Spectrum Scale configuration through
    - `mmaddnode`
    - `tdbdump` / `tdbrestore`



Putting it all together (3 of 3)

- Public sector customer @ United Kingdom
- Goal
  - Dynamically spin up / down self-contained application instances on AWS Cloud
  - Terminate instances EOB to save cost
- Challenges
  - Nodes join & leave clusters “at will”
  - Spectrum Scale doesn’t react well to dynamically changing IPs
- Solution
  - Two groups of EC2 instances:
    - Permanent (Quorum, NSD Servers, GUI)
    - Temporary (NSD Clients)
  - OS installation & configuration via custom AMIs
  - Spectrum Scale installation through Ansible
  - Spectrum Scale configuration through
    - systemd service units
    - `mmdelnode` / `mmaddnode`

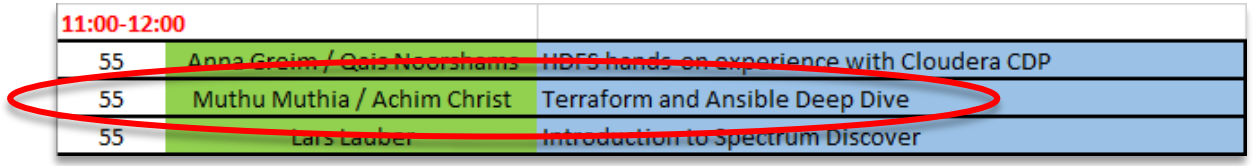


Summary

- Every automation use case is different
- Make it easy to deploy IBM Spectrum Scale
  - Integration points at various levels
  - Modular architecture (Lego bricks)
  - Open source, community-driven initiative
  - On-prem, public, hybrid – without vendor lock-in

- Learn more about Terraform & Ansible:
  - Benefits & Use Cases
  - Getting started with Ansible (& Spectrum Scale)
  - Getting started with Terraform (& Spectrum Scale)
  - Sample scenarios in more detail

→ *Terraform and Ansible Deep Dive*
Thu, 20th October, 11:00h



11:00-12:00		
55	Anna Greim / Qais Noorshams	HDFS hands-on experience with Cloudera CDP
55	Muthu Muthia / Achim Christ	Terraform and Ansible Deep Dive
55	Lars Lauber	Introduction to Spectrum Discover

Thank you for using
IBM Spectrum Scale!