Sabaas

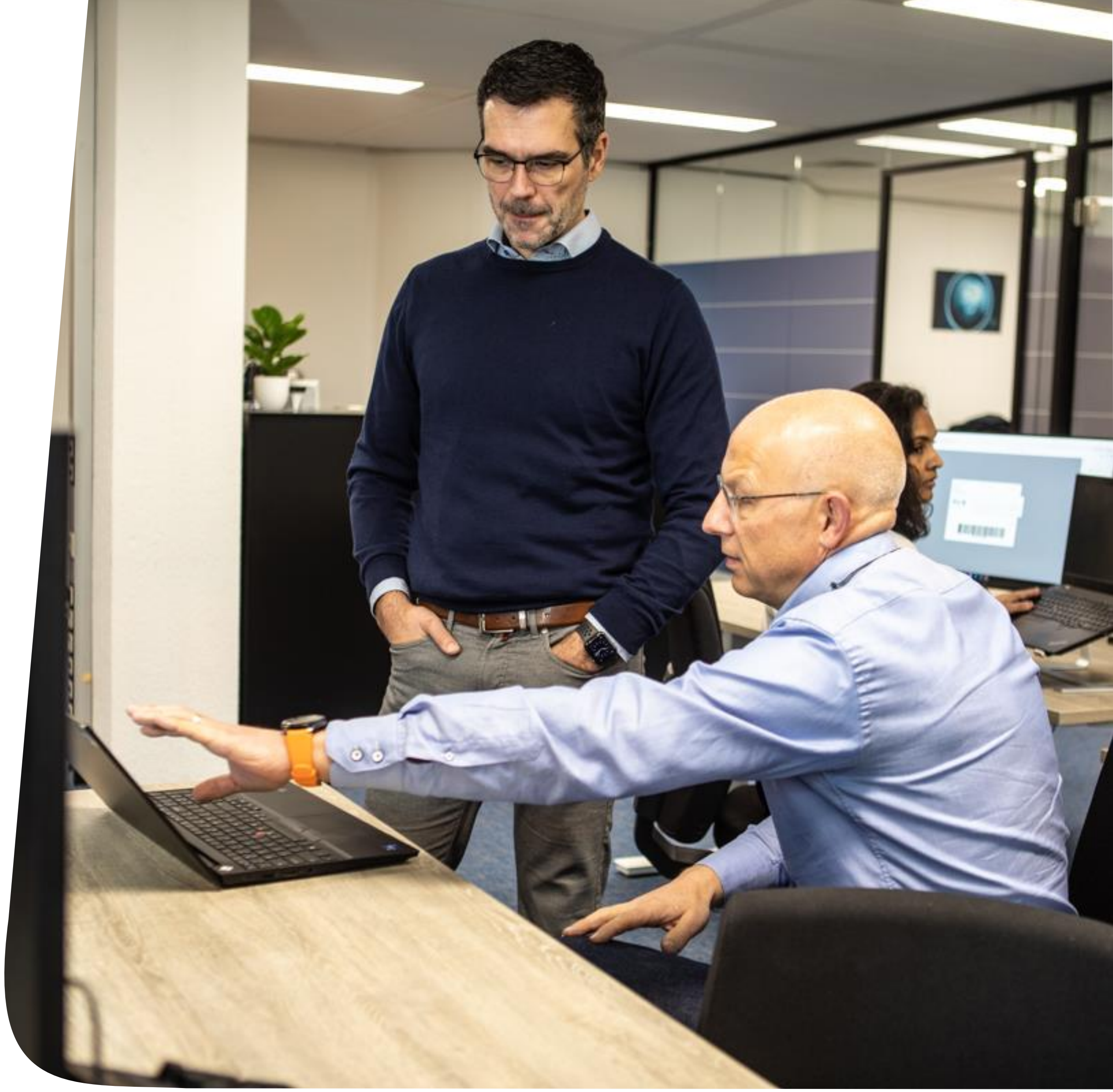
knowing change

SAP Basis on the move

Get ready for 2027

Agenda

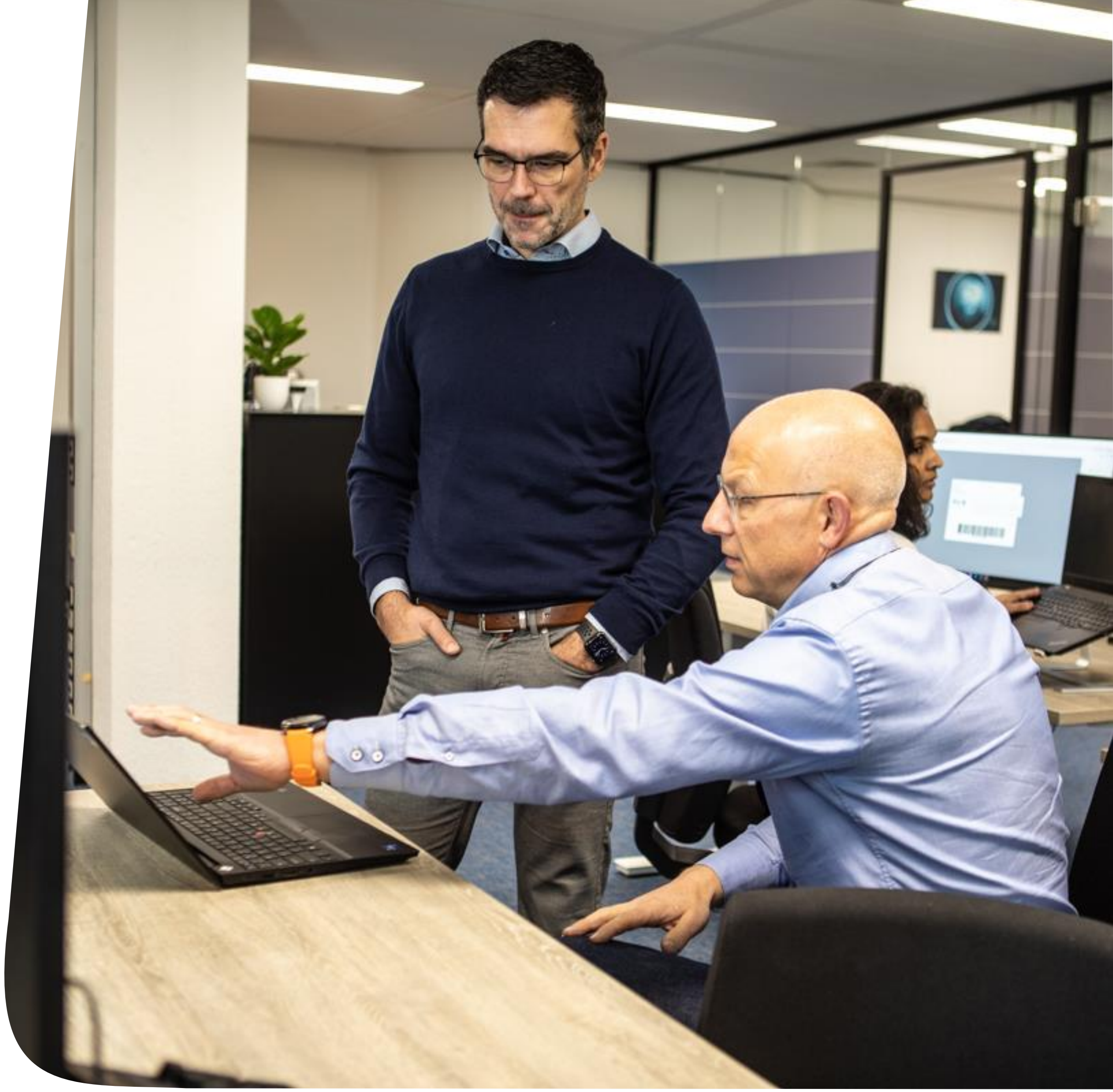
Introduction	18:00
SAP Basis: Where do we stand & where do we go	18:10
Break	19:15
SAP on AWS	19:30
Drinks	20:30



Sabaas
knowing change

Agenda

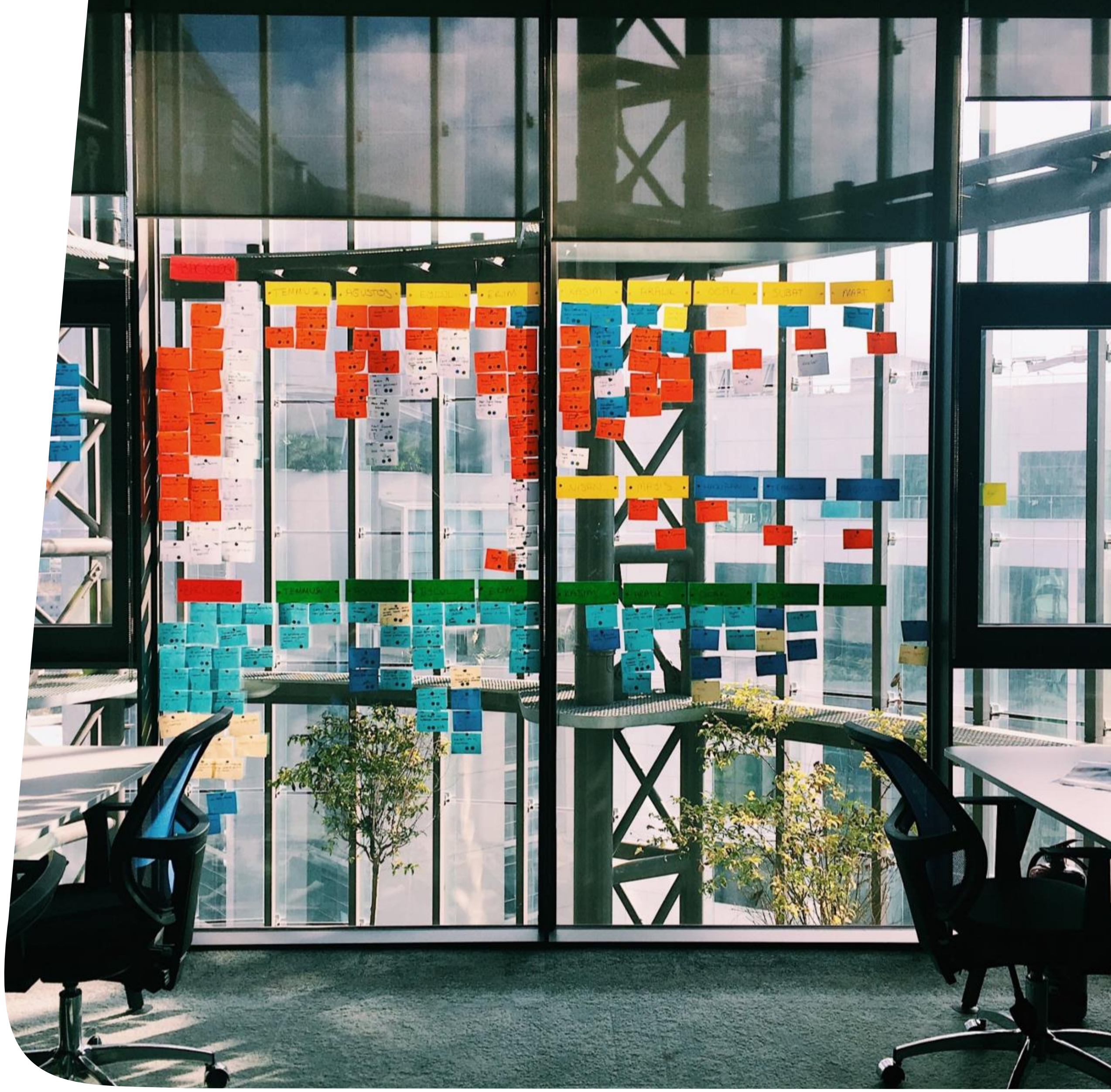
Introduction	18:00
SAP Basis: Where do we stand & where do we go	18:10
Break	19:15
SAP on AWS	19:30
Drinks	20:30



Sabaas
knowing change

Meeting notes

- Online / Offline
- Q&A
- Chat window Moderators



Sabaas
knowing change

Sabaas

Founded in 2015

SAP and Cloud Technical experts

ISO 27001 certified

AWS partner

Azure partner

Mitch van Dalfsen

Sales Director Sabaas | Knowing change

Sabaas
knowing change

in linkedin.com/in/mitchvandalfsen/

knowing change

We understand your organization is constantly changing. We know all the ins and outs of SAP and Cloud IT, and can answer all your questions regarding 'when', 'how' and 'why' to optimize your IT.

Sander Van Gemert

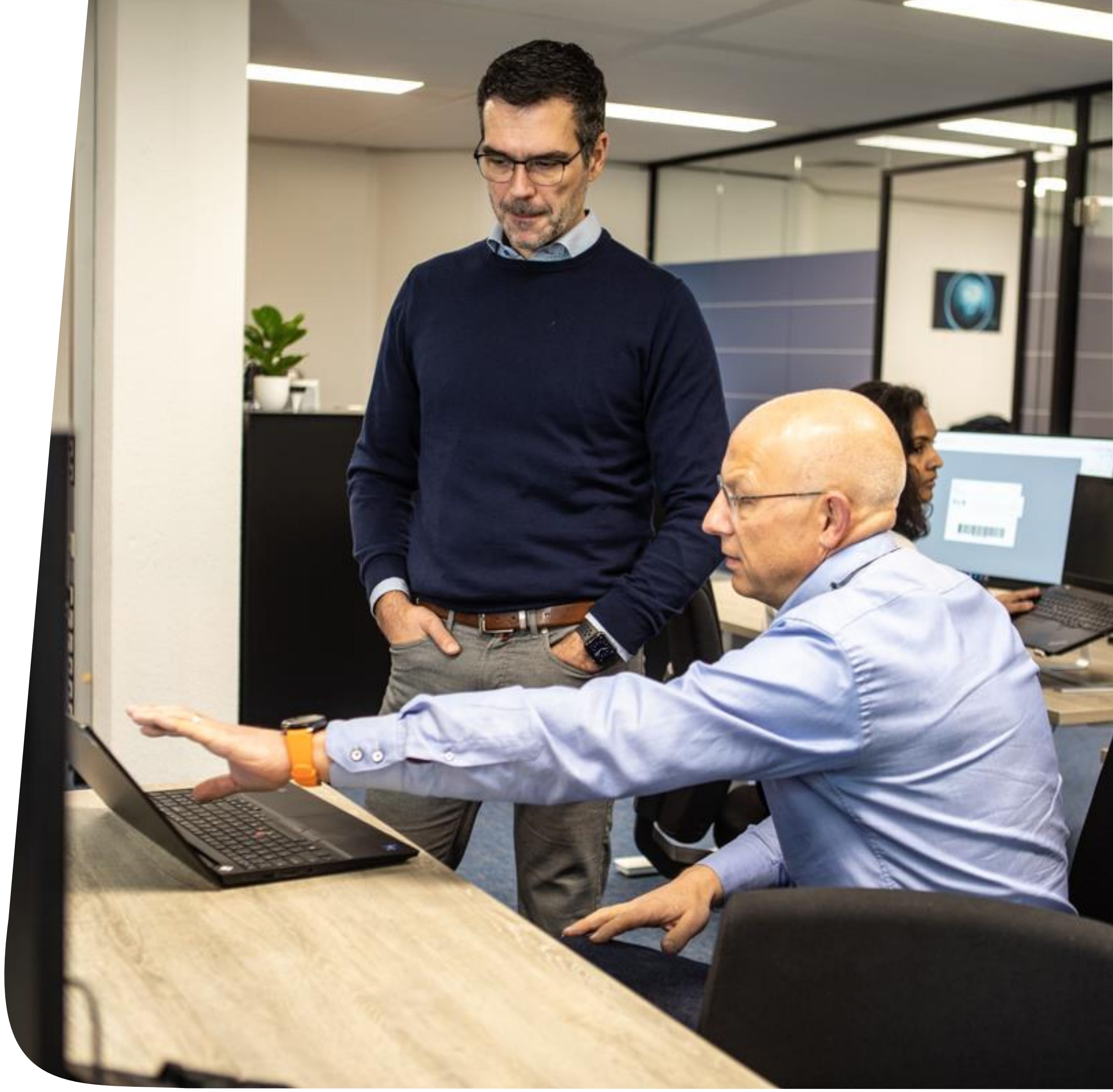
Managing Partner at Sabaas | Knowing change

Sabaas
knowing change

in linkedin.com/in/svgemert/

Agenda

Introduction	18:00
SAP Basis: Where do we stand & where do we go	18:10
Break	19:15
SAP on AWS	19:30
Drinks	20:30



Sabaas
knowing change

SAP Basis – 2019 Recap

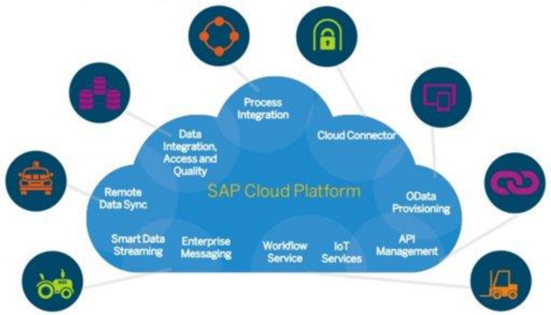
SAP S/4HANA

HashiCorp Terraform

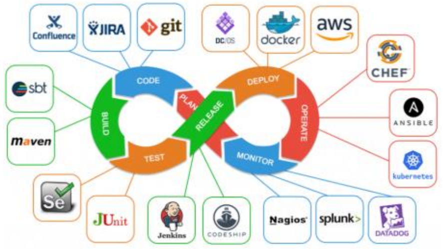
SAP 7.2 Solution Manager

CLOUD FOUNDRY

SAP Cloud Platform
Robust Portfolio of Integration Capabilities



SAP Business Warehouse



Google Cloud Platform

Microsoft Azure

aws

Sabaas
knowing change

Cloud native architecture

Cloud Native Principles

Pay as you go, afterwards
Self service—no waiting
Globally distributed by default
Cross-zone/region availability models
High utilization—turn idle resources off
Immutable code deployments

What changed

- SAP Cloud Platform -> SAP Business Technology platform
- HEC -> SAP RISE
- Solution Manager -> Cloud ALM
- SAP BW -> SAP Datawarehouse Cloud
- Sustainability
- Integration SAP data with non-SAP apps

Recap - Evolution of IT

Stage	**No process** 1990-2000	**Waterfall** 2000-2010	**Agile** 2010-2015	**Cloud Native** 2015-2025
Culture	Individual	Predictive	Iterative	Collaborative
Process	Random	Tightly planned	Scrum/Kanban	Cloud Native
Architecture	Emerging	Monolith	Service Oriented	Microservices
Monitoring	Re-active	Ad-hoc	Alerting	Self-healing
Delivery	Ad-hoc	Monthly releases	Continuous Integration	Continuous Delivery
Provisioning	Manual	Scripted	Configuration Management	Orchestration, infrastructure as Code
Infrastructure	Mainframe	Blades	Virtualization	Cloud
Security	User-id/password	VPN/token	Multi factor	Biometrics
			2019	**2022**

SAP Basis 2022

- SAP CORE untouched
- Focus on exposing SAP data
- New SAP implementations, on-premise, Cloud and RISE
- (Cloud)Migration and carve-out projects
- Service Integration and Management
- Data integrity, continuity and availability
- Automation with lowcode and Infra as Code

SAP Migrations

- Variations on migration
- RISE with SAP
- SAP operations
- Tools
- Key Challenges

Marc Kip

Consultant at Sabaas | Knowing change

linkedin.com/in/marc-kip/

Sabaas
knowing change

Variations on a Migration

SAP Datacenter Migration

- Host to host
- OS/DB Migration
- Carve out
- Data Migration

RISE with SAP

- Public Cloud
- Private Cloud

RISE with SAP

- Business Transformation
  - Lean vs. Complex
- Technology Platform
  - Hyperscaler ←
- Intelligent Digital Enterprise
  - Data driven

SAP Operations

Service Management

- CMDB
- Service Desk application
- Application monitoring
- Change and Release Mgmt

Integrated Access Mgmt

- User provisioning
- Single Sign On

Application

- Housekeeping and Job scheduling
- Interfacing
- Printers
- Handheld devices
- SAPGui/GuiXT/Persona/Web browsers
- Security
- Network topology and capacity
- Custom Code (ABAP/JAVA/...)
- NON-SAP Apps

Tools

- System copy
- DB log-file replication
- Transformation tools
  - EPI-USE
  - SNP
  - LTMC SAP (S/4HANA) / LSMW (ERP)
  - …
- Infrastructure & deployment tools
  - Ansible
  - Terraform

Key Challenges

Organizational / team

- Sending organization
- Receiving organization
- Suppliers
- Third Party specialists

Technical

- Tools (old but mature?)
- Network connectivity
- Sizing

Project Execution

- Test, test and test
- Team must understand process
- Flexibility
- Time

Business

- Minimize impact on operations
- Be consistent and predictable

SAP Infra as Code

- Traditional IT
- What is Infrastructure as Code (IaC)?
- IaC Tools
- What is Terraform?
- Terraform vs Ansible
- What is BitBucket?
- How does Terraform work?
- How does Ansible work?

Sabaas
knowing change

Neety Sharma

SAP Technology Consultant at

Sabaas | Knowing change

Neha Puthran

SAP Technology Consultant at

Sabaas | Knowing change

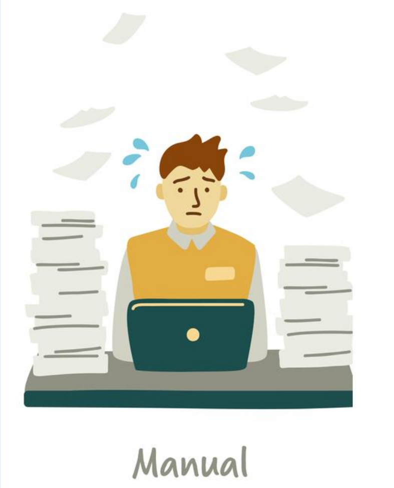
Traditional IT

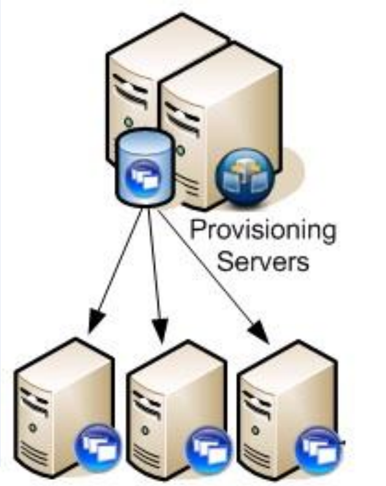
Setup servers

- Configure network and route tables
- Create install software
- Install database and application

Maintain servers

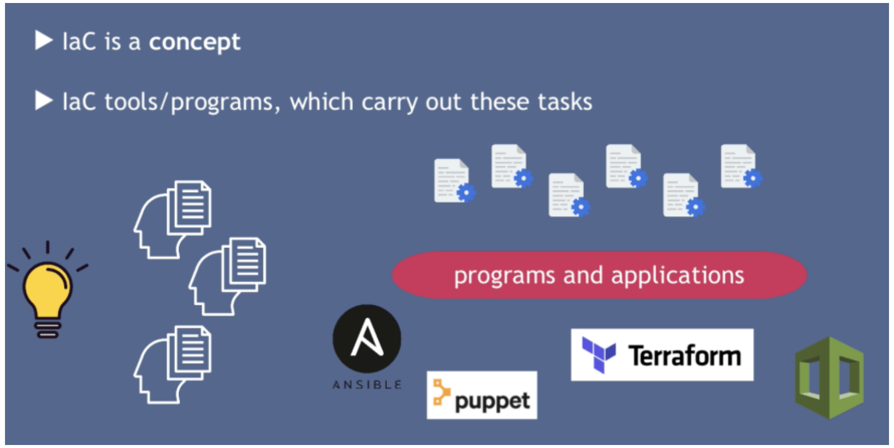
- Update system version
- Update OS packages
- Database backups
- Same process for multiple environments





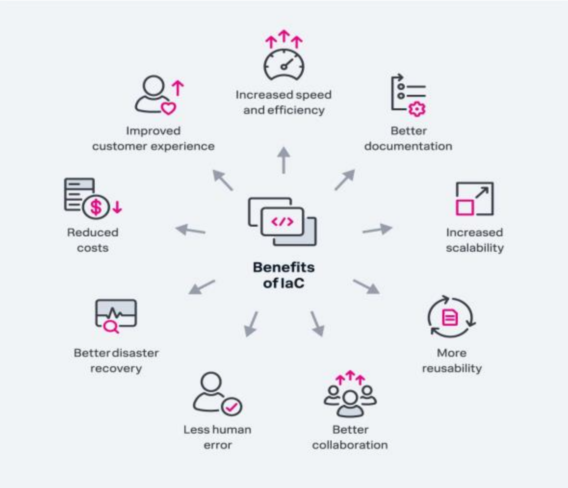
What is Infrastructure as Code (IaC)?

- Configures infrastructure through descriptive model
- Treats infrastructure like application source code
- Configuration stored in version control systems
- Provides greater accuracy, reduced errors, increased speed and consistency



▶ IaC is a concept

▶ IaC tools/programs, which carry out these tasks



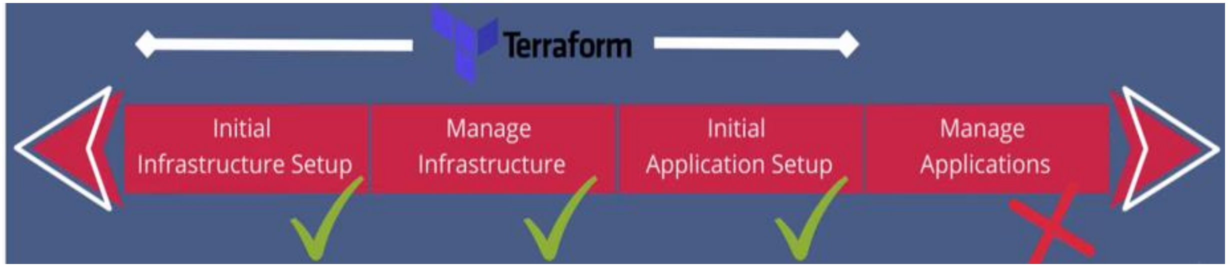
Categories of IaC tools:

- Infrastructure provisioning
- Configuration of provisioned infrastructure
- Application deployment

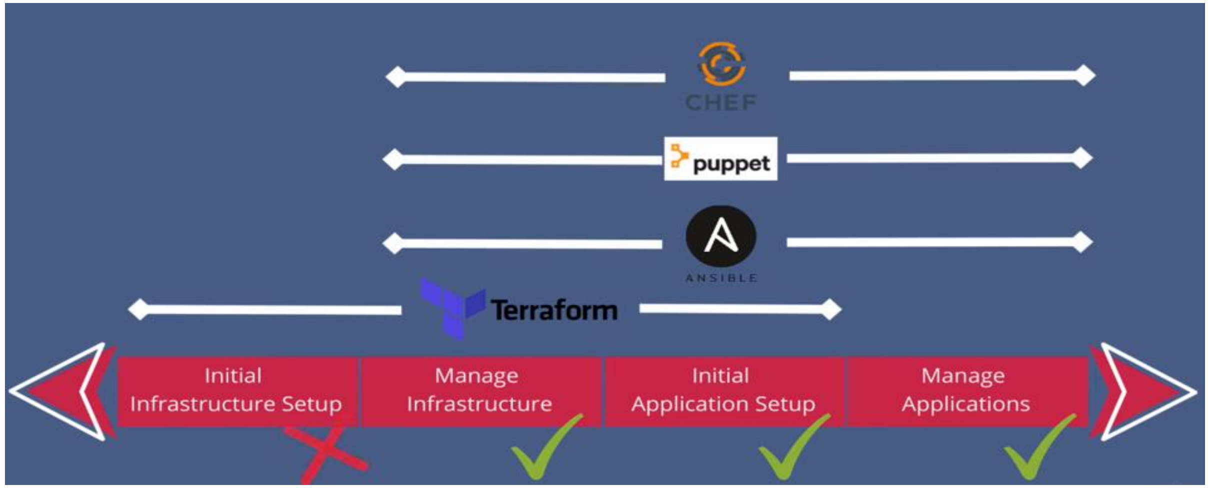
Why do we have so many IaC tools?

- The same tool cannot perform all tasks
- Each tool is used for specific phase

Initial Setup Phase



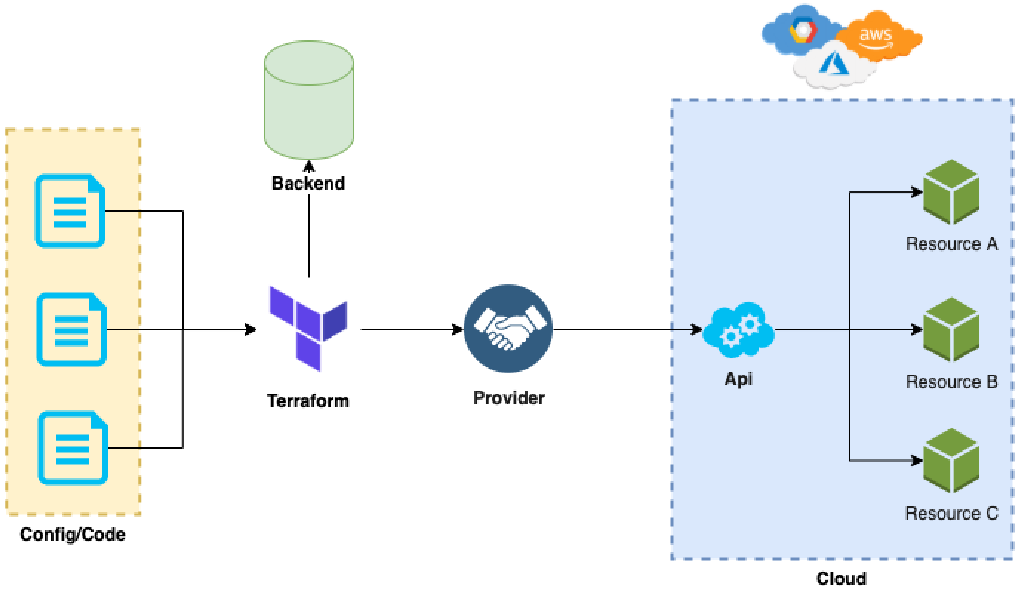
Maintaining Phase

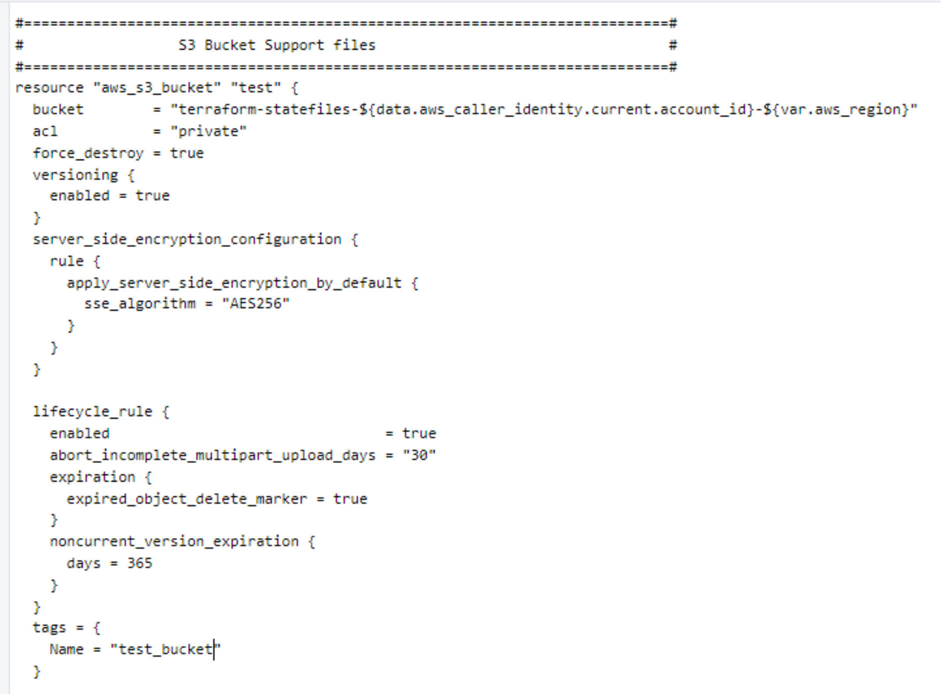


Sabaas
knowing change

What is Terraform?

- Automates and manages
  - your infrastructure
  - your platform
  - and services which run on that platform
- Open source
- Declarative = define the result you want
- Tool for infrastructure provisioning



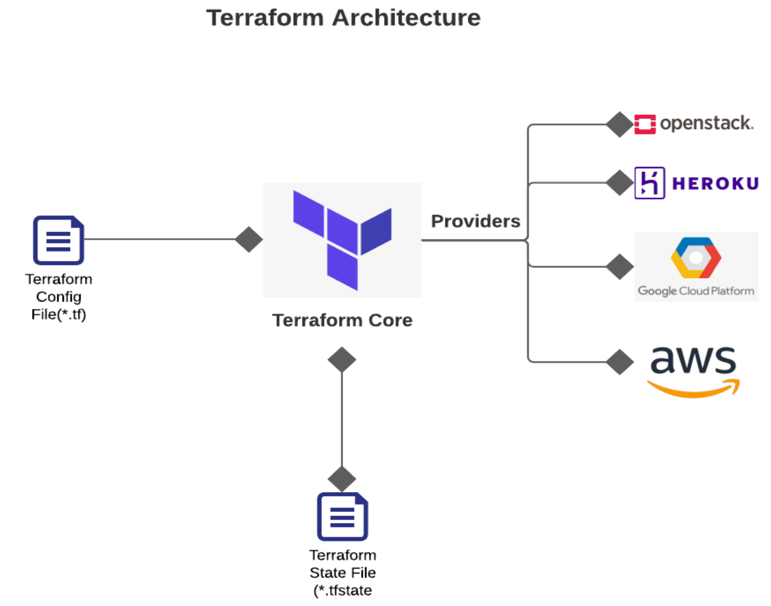


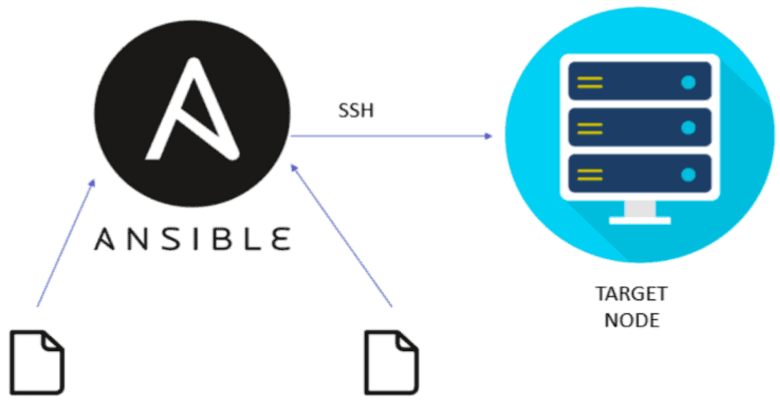
```
#============================================================================#
#                  S3 Bucket Support files                                   #
#============================================================================#
resource "aws_s3_bucket" "test" {
  bucket        = "terraform-statefiles-${data.aws_caller_identity.current.account_id}-${var.aws_region}"
  acl           = "private"
  force_destroy = true
  versioning {
    enabled = true
  }
  server_side_encryption_configuration {
    rule {
      apply_server_side_encryption_by_default {
        sse_algorithm = "AES256"
      }
    }
  }

  lifecycle_rule {
    enabled                                = true
    abort_incomplete_multipart_upload_days = "30"
    expiration {
      expired_object_delete_marker = true
    }
    noncurrent_version_expiration {
      days = 365
    }
  }
  tags = {
    Name = "test_bucket"
  }
```

Sabaas
knowing change

Terraform vs Ansible

Factor	Terraform	Ansible
Type	Orchestration tool	Configuration management tool
Language	Declarative	Procedural
VM Provisioning, networking & storage management	Provides comprehensive VM provisioning, networking & storage management	Provides partial VM provisioning, networking & storage management

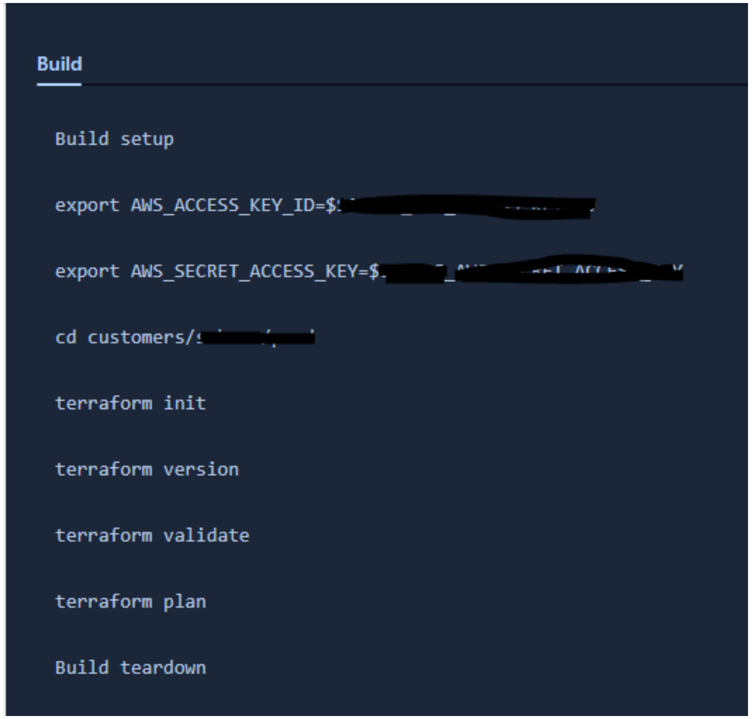




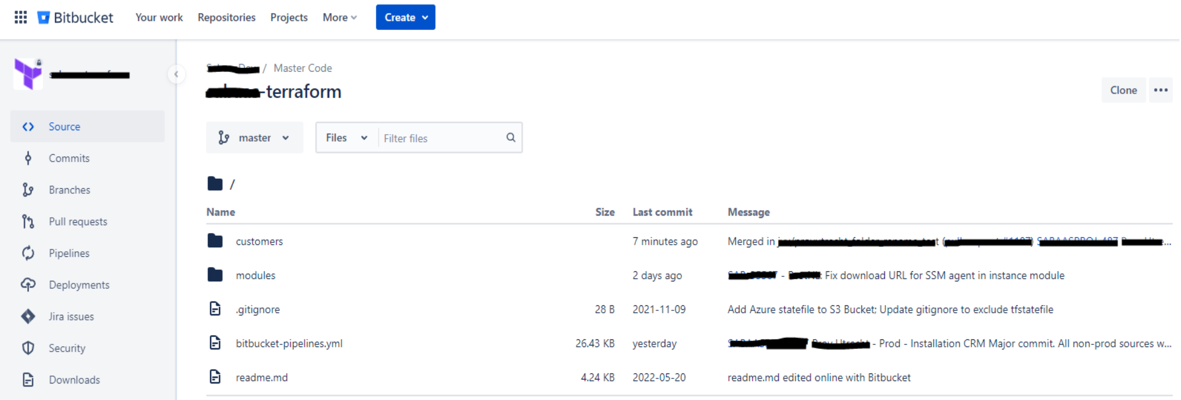
Sabaas
knowing change

What is Bitbucket?

- Git repository management solution designed for professional teams
- Access control to restrict access to your source code
- Workflow control to enforce team workflow
- Pull requests with in-line commenting for collaboration on code review



```
Build setup

export AWS_ACCESS_KEY_ID=$[redacted]

export AWS_SECRET_ACCESS_KEY=$[redacted]

cd customers/[redacted]

terraform init

terraform version

terraform validate

terraform plan

Build teardown
```



[redacted] / Master Code

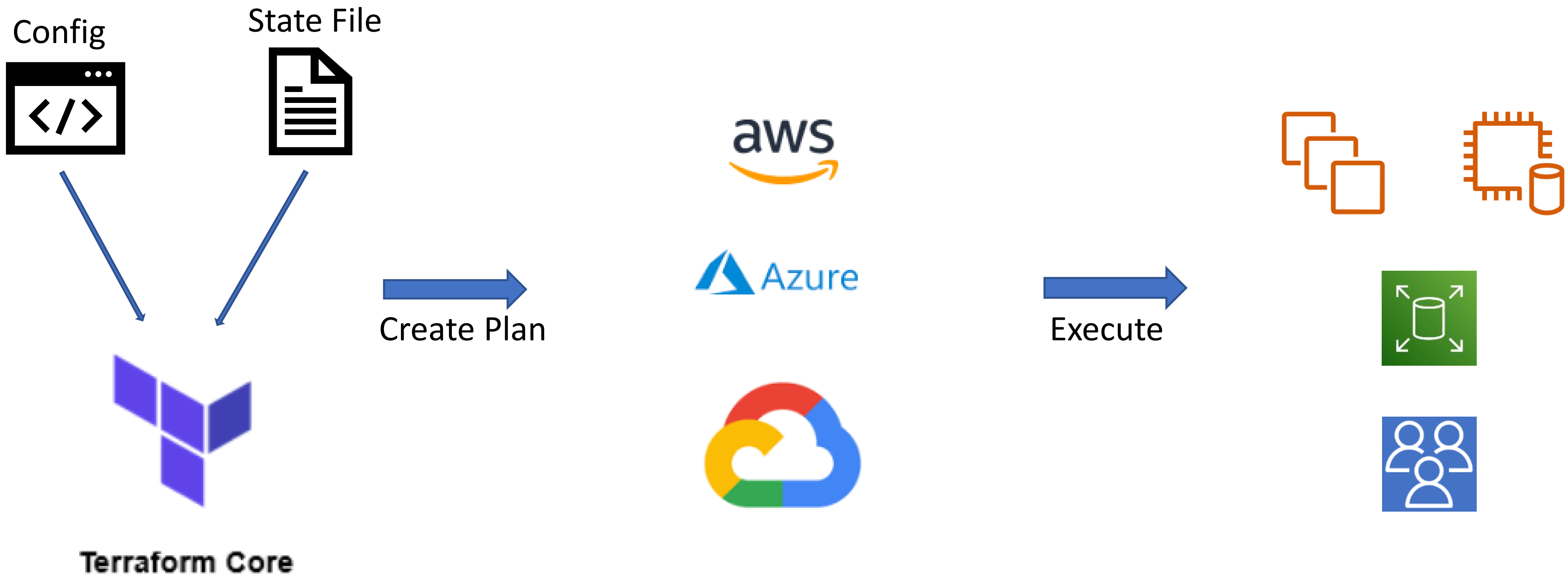
[redacted]-terraform

/

Name	Size	Last commit	Message
customers		7 minutes ago	Merged in [redacted]...
modules		2 days ago	[redacted] Fix download URL for SSM agent in instance module
.gitignore	28 B	2021-11-09	Add Azure statefile to S3 Bucket; Update gitignore to exclude tfstatefile
bitbucket-pipelines.yml	26.43 KB	yesterday	[redacted] - Prod - Installation CRM Major commit. All non-prod sources w...
readme.md	4.24 KB	2022-05-20	readme.md edited online with Bitbucket

Sabaas
knowing change

How does Terraform work?



Config

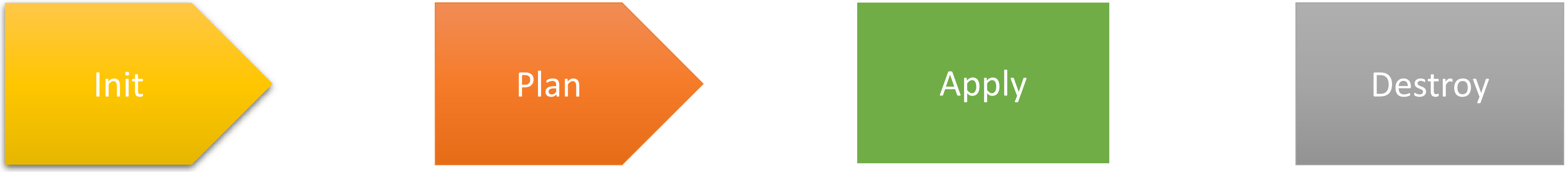
State File

Terraform Core

Create Plan

Execute

Terraform Commands



Init

Plan

Apply

Destroy

terraform init

Initializes terraform before the plan execution eg. checking provider access keys, checking working directory

terraform plan

Creates a plan for the desired state as per the changes in the config file

terraform apply

Performs the actual execution of the plan

terraform destroy

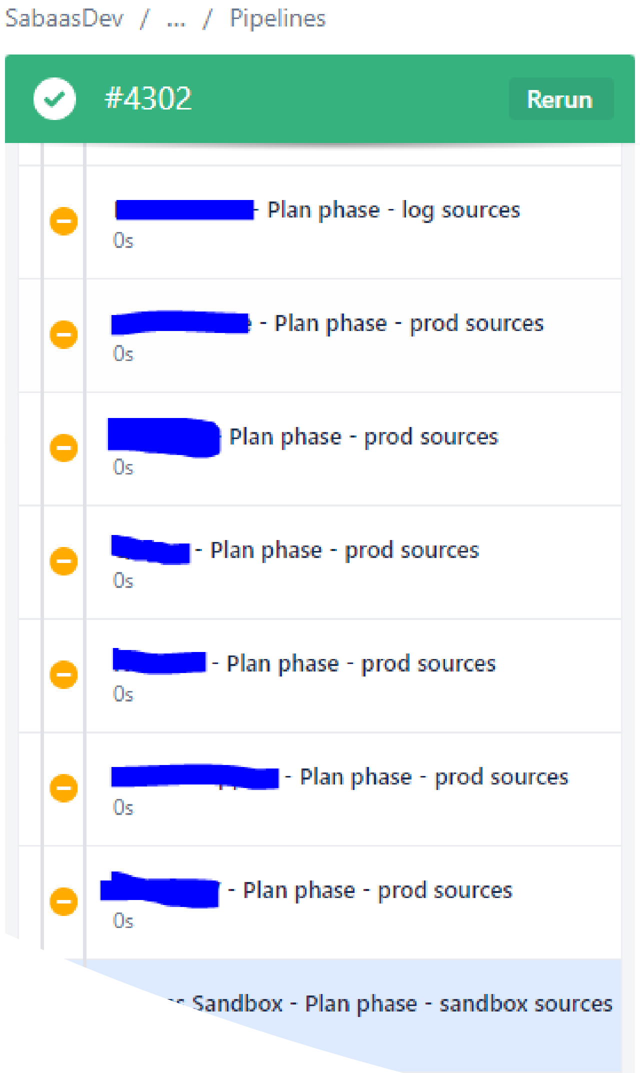
Removes elements 1 by 1 and in an order.

Example for Terraform

- Create an AWS EC2 instance named demo
- Will use the Amazon machine Image(AMI) as specified for creating this instance
- The instance created will be of type t2.micro

```
resource "aws_instance" "demo"

{

ami = "ami-06bc7889ee68f279e"

instance_type = "t2.micro"

}
```

Sabaas
knowing change



SabaasDev / ... / Pipelines

#4302 Rerun

[redacted] Plan phase - log sources
0s

[redacted] - Plan phase - prod sources
0s

[redacted] Plan phase - prod sources
0s

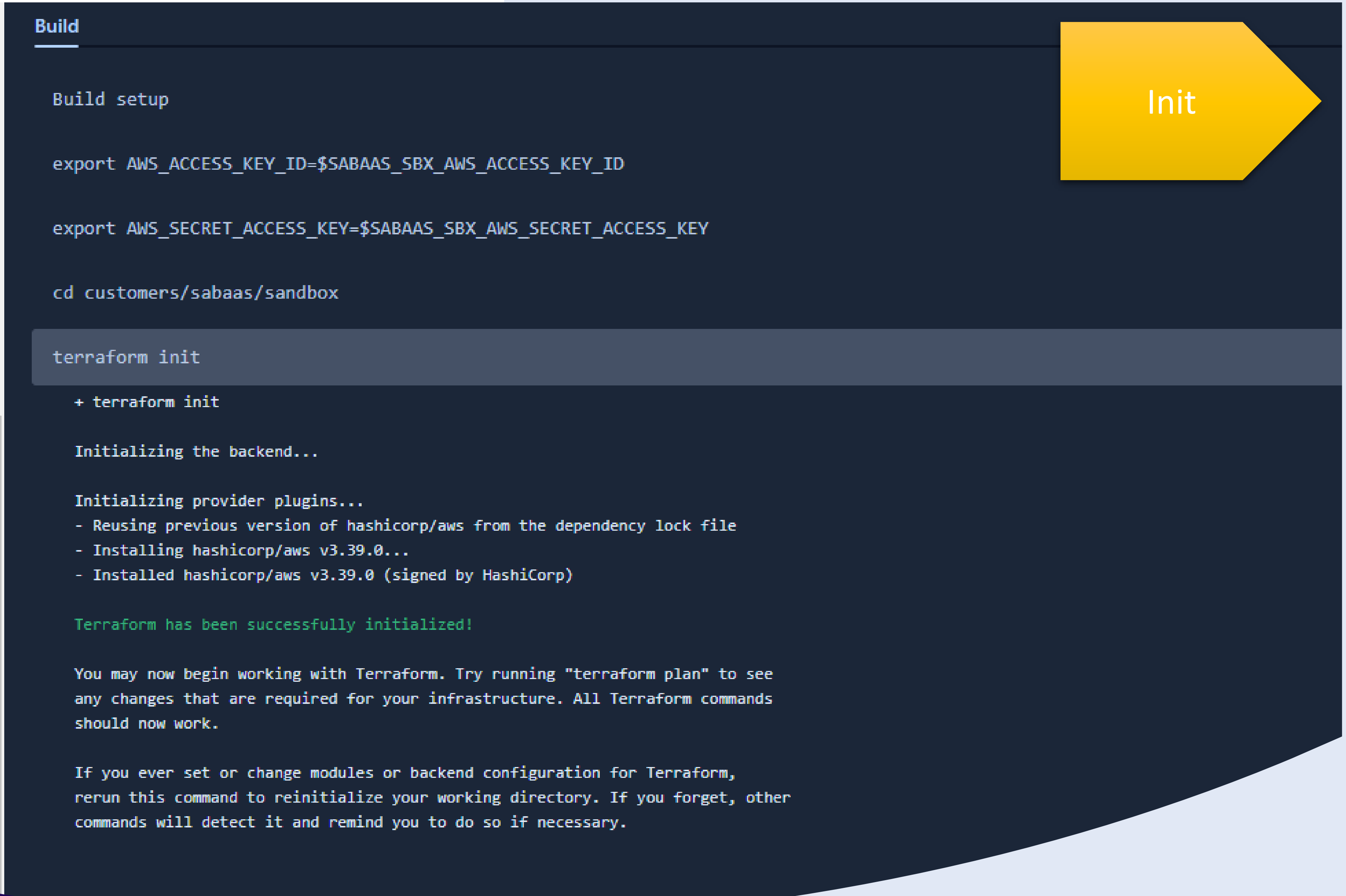
[redacted] - Plan phase - prod sources
0s

[redacted] - Plan phase - prod sources
0s

[redacted] - Plan phase - prod sources
0s

[redacted] - Plan phase - prod sources
0s

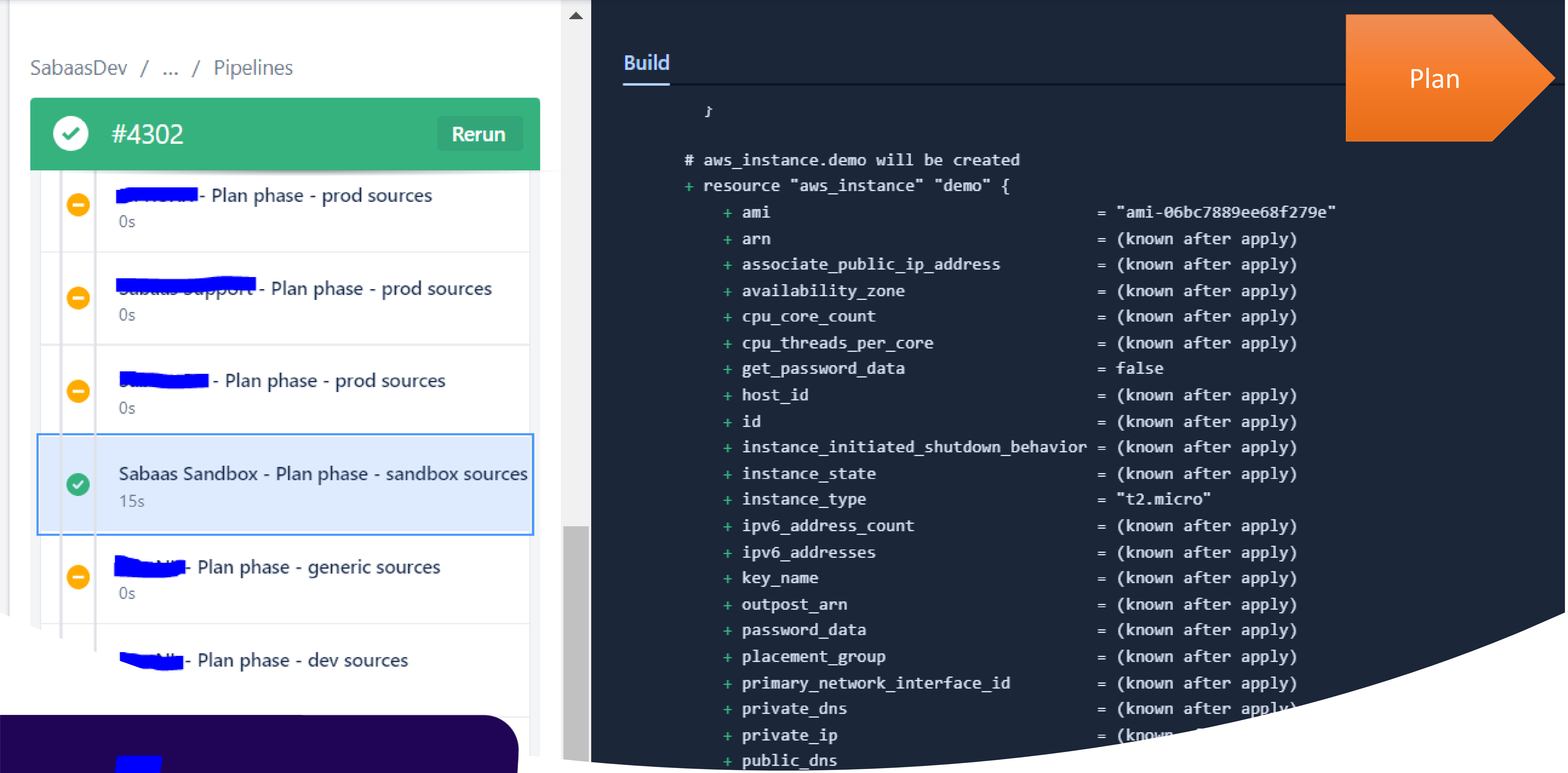
Sandbox - Plan phase - sandbox sources



```
Build setup

export AWS_ACCESS_KEY_ID=$SABAAS_SBX_AWS_ACCESS_KEY_ID

export AWS_SECRET_ACCESS_KEY=$SABAAS_SBX_AWS_SECRET_ACCESS_KEY

cd customers/sabaas/sandbox

terraform init
  + terraform init

  Initializing the backend...

  Initializing provider plugins...
  - Reusing previous version of hashicorp/aws from the dependency lock file
  - Installing hashicorp/aws v3.39.0...
  - Installed hashicorp/aws v3.39.0 (signed by HashiCorp)

  Terraform has been successfully initialized!

  You may now begin working with Terraform. Try running "terraform plan" to see
  any changes that are required for your infrastructure. All Terraform commands
  should now work.

  If you ever set or change modules or backend configuration for Terraform,
  rerun this command to reinitialize your working directory. If you forget, other
  commands will detect it and remind you to do so if necessary.
```

Init

Sabaas
knowing change



SabaasDev / ... / Pipelines

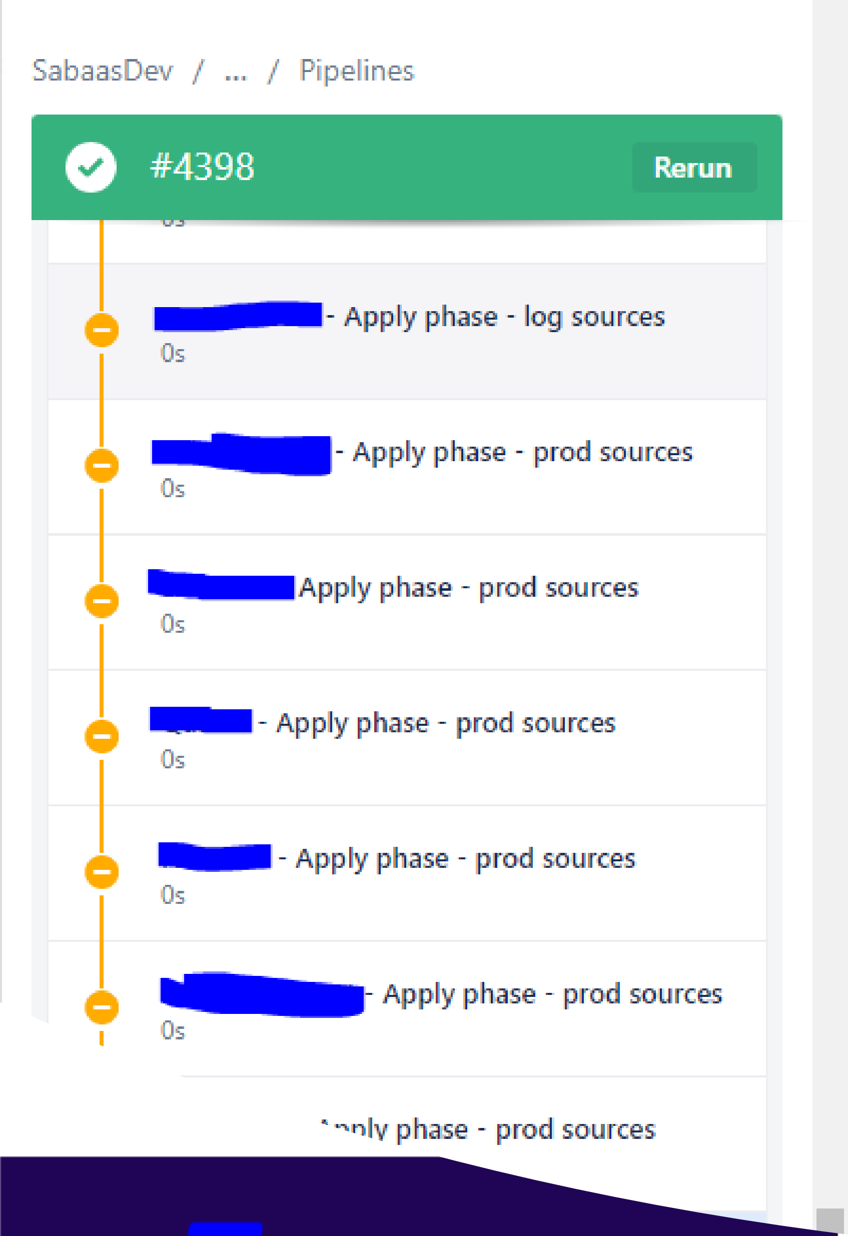
#4302 Rerun

- [redacted] - Plan phase - prod sources
  0s
- [redacted] - Plan phase - prod sources
  0s
- [redacted] - Plan phase - prod sources
  0s
- Sabaas Sandbox - Plan phase - sandbox sources
  15s
- [redacted] - Plan phase - generic sources
  0s
- [redacted] - Plan phase - dev sources

Build

```
        }

  # aws_instance.demo will be created
  + resource "aws_instance" "demo" {
      + ami                                  = "ami-06bc7889ee68f279e"
      + arn                                  = (known after apply)
      + associate_public_ip_address          = (known after apply)
      + availability_zone                    = (known after apply)
      + cpu_core_count                       = (known after apply)
      + cpu_threads_per_core                 = (known after apply)
      + get_password_data                    = false
      + host_id                              = (known after apply)
      + id                                   = (known after apply)
      + instance_initiated_shutdown_behavior = (known after apply)
      + instance_state                       = (known after apply)
      + instance_type                        = "t2.micro"
      + ipv6_address_count                   = (known after apply)
      + ipv6_addresses                       = (known after apply)
      + key_name                             = (known after apply)
      + outpost_arn                          = (known after apply)
      + password_data                        = (known after apply)
      + placement_group                      = (known after apply)
      + primary_network_interface_id         = (known after apply)
      + private_dns                          = (known after apply)
      + private_ip                           = (known
      + public_dns
```

Sabaas
knowing change



SabaasDev / ... / Pipelines

✓ #4398 Rerun

[redacted] - Apply phase - log sources
0s

[redacted] - Apply phase - prod sources
0s

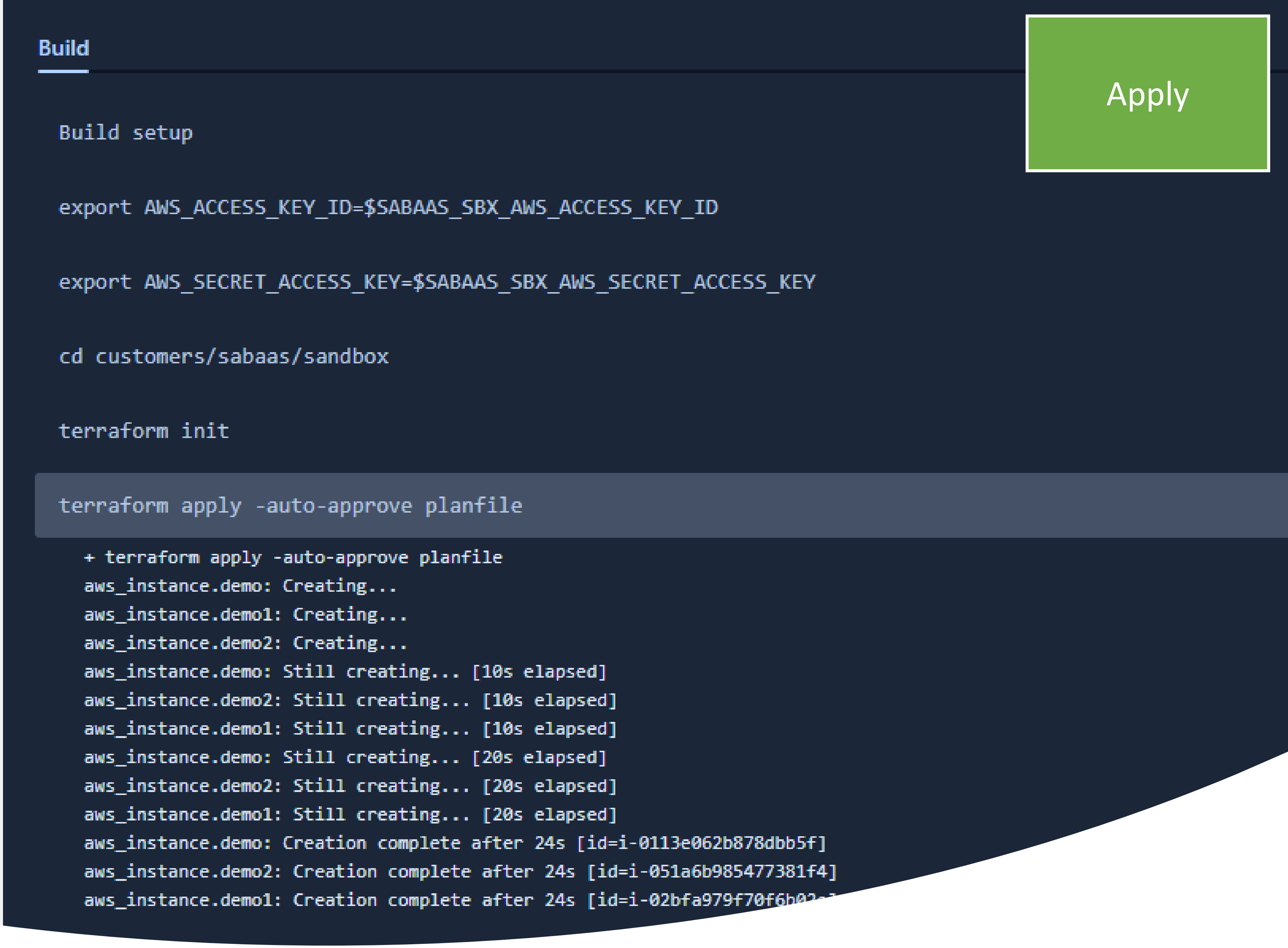
[redacted] Apply phase - prod sources
0s

[redacted] - Apply phase - prod sources
0s

[redacted] - Apply phase - prod sources
0s

[redacted] - Apply phase - prod sources
0s

pply phase - prod sources



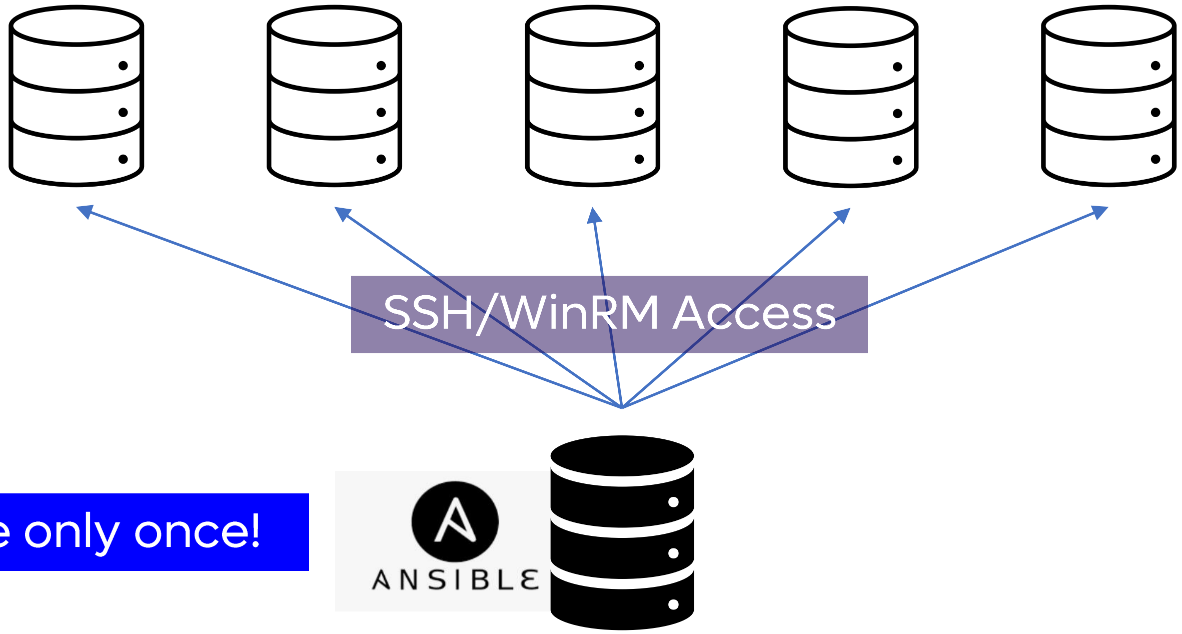
Build

```
Build setup

export AWS_ACCESS_KEY_ID=$SABAAS_SBX_AWS_ACCESS_KEY_ID

export AWS_SECRET_ACCESS_KEY=$SABAAS_SBX_AWS_SECRET_ACCESS_KEY

cd customers/sabaas/sandbox

terraform init

terraform apply -auto-approve planfile

  + terraform apply -auto-approve planfile
  aws_instance.demo: Creating...
  aws_instance.demo1: Creating...
  aws_instance.demo2: Creating...
  aws_instance.demo: Still creating... [10s elapsed]
  aws_instance.demo2: Still creating... [10s elapsed]
  aws_instance.demo1: Still creating... [10s elapsed]
  aws_instance.demo: Still creating... [20s elapsed]
  aws_instance.demo2: Still creating... [20s elapsed]
  aws_instance.demo1: Still creating... [20s elapsed]
  aws_instance.demo: Creation complete after 24s [id=i-0113e062b878dbb5f]
  aws_instance.demo2: Creation complete after 24s [id=i-051a6b985477381f4]
  aws_instance.demo1: Creation complete after 24s [id=i-02bfa979f70f6b02
```

Apply

Sabaas
knowing change

Ansible



Install ansible only once!

Control server

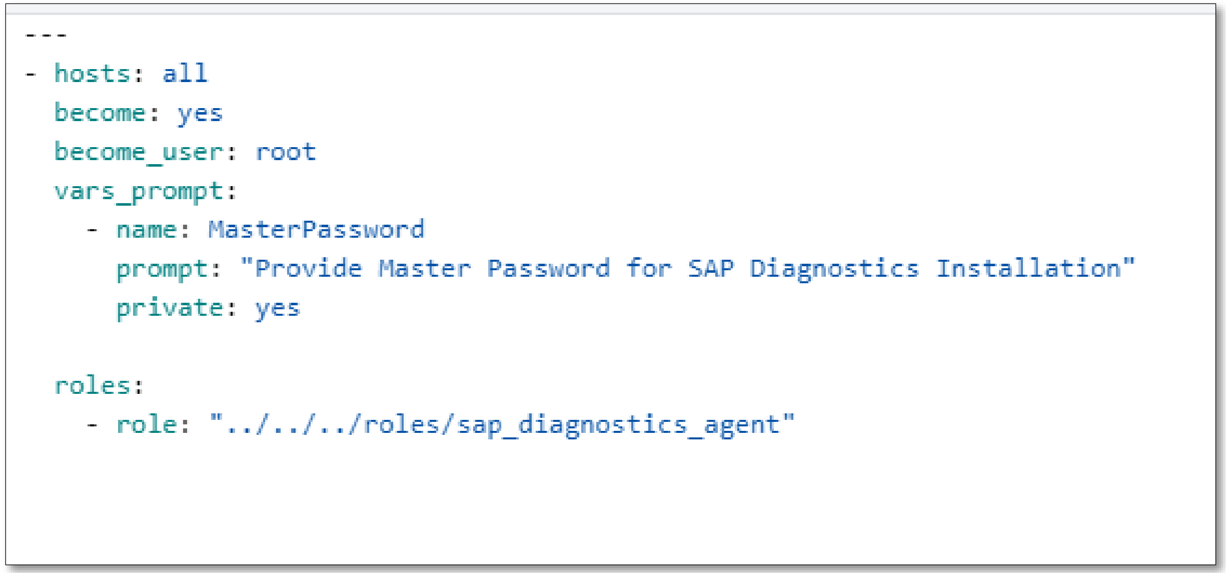
Ansible Structure

Roles Made from small specific tasks modules

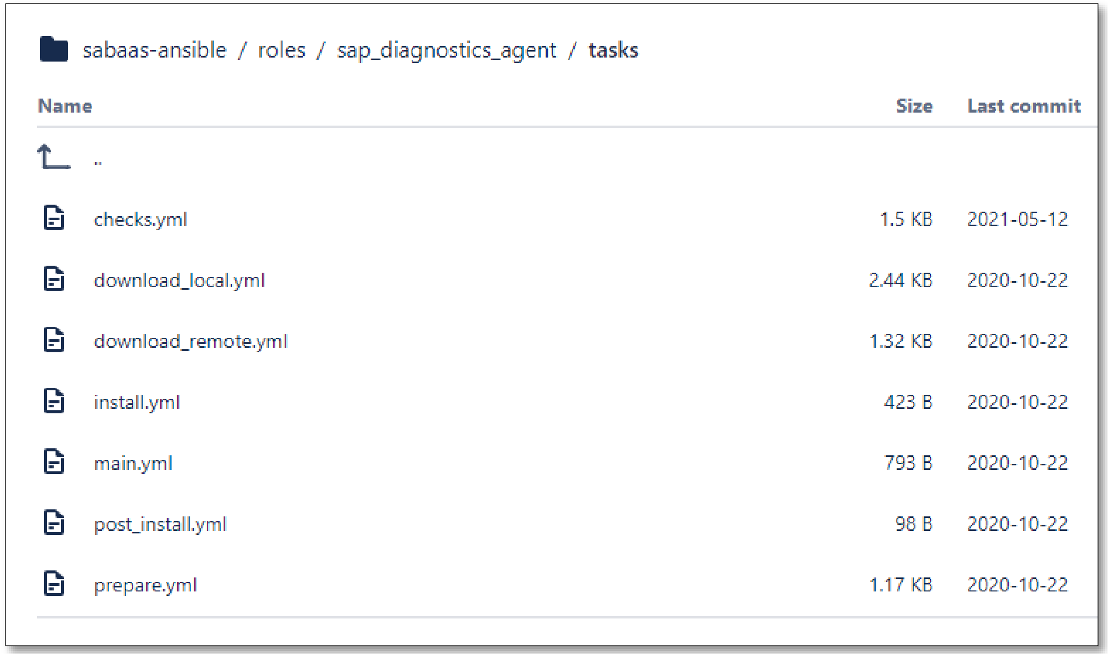
Playbooks A group of roles to accomplish a complete configuration

Inventory A list of all the servers where the playbooks will be executed

Example Ansible



```yaml
---
- hosts: all
  become: yes
  become_user: root
  vars_prompt:
    - name: MasterPassword
      prompt: "Provide Master Password for SAP Diagnostics Installation"
      private: yes

  roles:
    - role: "../../../roles/sap_diagnostics_agent"
```



sabaas-ansible / roles / sap_diagnostics_agent / **tasks**

Name	Size	Last commit
..		
checks.yml	1.5 KB	2021-05-12
download_local.yml	2.44 KB	2020-10-22
download_remote.yml	1.32 KB	2020-10-22
install.yml	423 B	2020-10-22
main.yml	793 B	2020-10-22
post_install.yml	98 B	2020-10-22
prepare.yml	1.17 KB	2020-10-22

Sabaas
knowing change

Many possibilities

Apply patches and OS upgrades using Terraform

Terraform to build new system, manage the existing ones

Terraform to provision AWS resources, manage them

Perform setup of new systems and configuration via Ansible

Keep track of changes and versioning of the developed and configured code

… And the list by no means complete!

Final Takeaway

- Speed
- Reduced risk
- Configuration consistency
- Documentation
- Enhanced security

Let the machine do the repetitive tasks…
while we INNOVATE!

Sabaas
knowing change

Sabaas

knowing change

Example Demo

Agenda

Introduction	18:00
SAP Basis: Where do we stand & where do we go	18:10
Break	19:15
SAP on AWS	19:30
Drinks	20:30

Sabaas
knowing change

Agenda

Introduction	18:00
SAP Basis: Where do we stand & where do we go	18:10
Break	19:15
SAP on AWS	19:30
Drinks	20:30

Agenda

Introduction	18:00
SAP Basis: Where do we stand & where do we go	18:10
Break	19:15
SAP on AWS	19:30
Drinks	**20:30**

Sabaas
knowing change