

Ansible for z/OS

—

GSE z/OS zExpertenForum
Vitznau (CH), 20. Oktober 2021

Jürgen Holtz
holtz@de.ibm.com

Automating your Z Environment with Ansible

—

Dan Jast

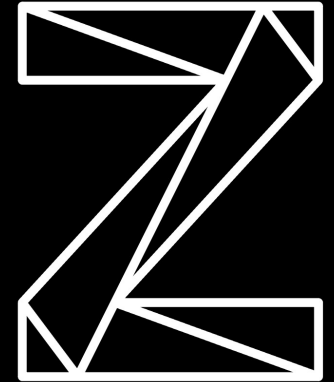
Product Manager - Red Hat Ansible Certified Content
for IBM Z

daniel.jast@ibm.com

Bryant Panyarachun

Development Lead – Red Hat Ansible Certified Content
for IBM Z

bpanyar@us.ibm.com



IBM

Please note

IBM's statements regarding its plans, directions, and intent are subject to change or withdrawal without notice and 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. Information about potential future products may not be incorporated into any contract.

The development, release, and timing of any future features or functionality described for our products remains at our sole discretion.

Performance is based on measurements and projections using standard IBM benchmarks in a controlled environment. The actual throughput or performance that any user will experience will vary depending upon many factors, including considerations such as the amount of multiprogramming in the user's job stream, the I/O configuration, the storage configuration, and the workload processed. Therefore, no assurance can be given that an individual user will achieve results similar to those stated here.

Notices and disclaimers

© 2021 International Business Machines Corporation. No part of this document may be reproduced or transmitted in any form without written permission from IBM.

U.S. Government Users Restricted Rights – use, duplication or disclosure restricted by GSA ADP Schedule Contract with IBM.

This document is current as of the initial date of publication and may be changed by IBM at any time. Not all offerings are available in every country in which IBM operates.

Information in these presentations (including information relating to products that have not yet been announced by IBM) has been reviewed for accuracy as of the date of initial publication and could include unintentional technical or typographical errors. IBM shall have no responsibility to update this information. **This document is distributed “as is” without any warranty, either express or implied. In no event, shall IBM be liable for any damage arising from the use of this information, including but not limited to, loss of data, business interruption, loss of profit or loss of opportunity.** IBM products and services are warranted per the terms and conditions of the agreements under which they are provided. The performance data and client examples cited are presented for illustrative purposes only. Actual performance results may vary depending on specific configurations and operating conditions.

IBM products are manufactured from new parts or new and used parts. In some cases, a product may not be new and may have been previously installed. Regardless, our warranty terms apply.”

Any statements regarding IBM's future direction, intent or product plans are subject to change or withdrawal without notice.

Performance data contained herein was generally obtained in a controlled, isolated environments. Customer examples are presented as illustrations of how those customers have used IBM products and the results they may have achieved. Actual performance, cost, savings or other results in other operating environments may vary.

References in this document to IBM products, programs, or services does not imply that IBM intends to make such products, programs or services available in all countries in which IBM operates or does business.

Workshops, sessions and associated materials may have been prepared by independent session speakers, and do not necessarily reflect the views of IBM. All materials and discussions are provided for informational purposes only, and are neither intended to, nor shall constitute legal or other guidance or advice to any individual participant or their specific situation.

Notices and disclaimers continued

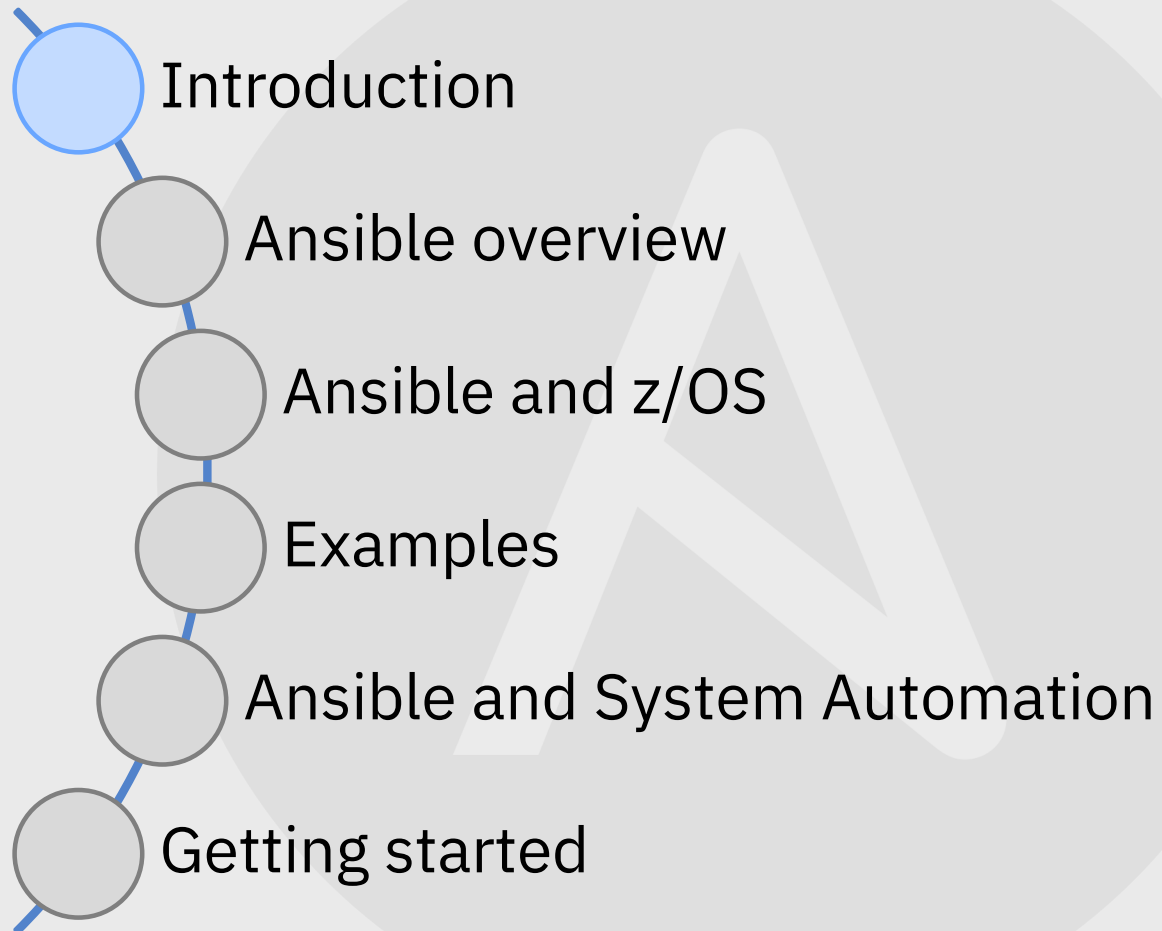
It is the customer's responsibility to insure its own compliance with legal requirements and to obtain advice of competent legal counsel as to the identification and interpretation of any relevant laws and regulatory requirements that may affect the customer's business and any actions the customer may need to take to comply with such laws. IBM does not provide legal advice or represent or warrant that its services or products will ensure that the customer follows any law.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products about this publication and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products. IBM does not warrant the quality of any third-party products, or the ability of any such third-party products to interoperate with IBM's products. **IBM expressly disclaims all warranties, expressed or implied, including but not limited to, the implied warranties of merchantability and fitness for a purpose.**

The provision of the information contained herein is not intended to, and does not, grant any right or license under any IBM patents, copyrights, trademarks or other intellectual property right.

IBM, the IBM logo, and ibm.com are trademarks of International Business Machines Corporation, registered in many jurisdictions worldwide. Other product and service names might be trademarks of IBM or other companies. A current list of IBM trademarks is available on the Web at "Copyright and trademark information" at: www.ibm.com/legal/copytrade.shtml.

Agenda



Ansible - Overview

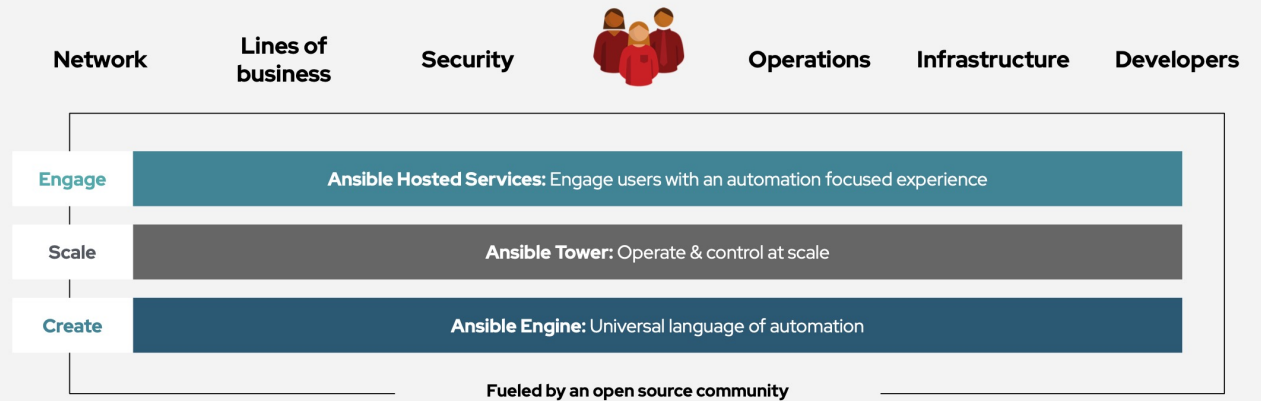
RED HAT ANSIBLE AUTOMATION - INTRODUCING THE AUTOMATION PLATFORM

Ansible Automation is the enterprise **framework** for automating across IT operations.

Ansible Engine[®] runs Ansible Playbooks, the automation **language** that can perfectly describe an IT application infrastructure.

Ansible Tower[®] allows you **scale** IT automation, manage complex deployments and speed productivity.

Red Hat Ansible Automation Platform



Red Hat Ansible Automation Platform



Flexibility

Bring disparate IT into a coherent whole using a market leading open solution backed with enterprise support

Interact directly with z/OS resources or integrate with existing platform tools



Consistency

Integrate z/OS into an enterprise automation strategy in a consistent way

Centralize management of your IT infrastructure



Simplicity

The certified collections codify much of the z/OS specific knowledge and complexity

Developer or system programmer can focus on their tasks and be more productive

Ansible managing to z/OS

Key Client Use Cases



Provisioning and Maintenance

- Build and provision middleware
- Roll out fix packs to thousand of servers
- Self-service provisioning portals



Configuration management

- Middleware configuration
- Network and security configuration



Security Automation

- SSL certification renewal process
- Password resets, create new users



CI/CD and Application Deployment

- Integrate infrastructure provisioning and Z application deployment into CI/CD pipeline



Orchestration

- Orchestrate and replace existing siloed in-house automation
- Integrate existing automation into overall workflow



Probe the mainframe

- Collect audit and security configuration details, system status and health checks

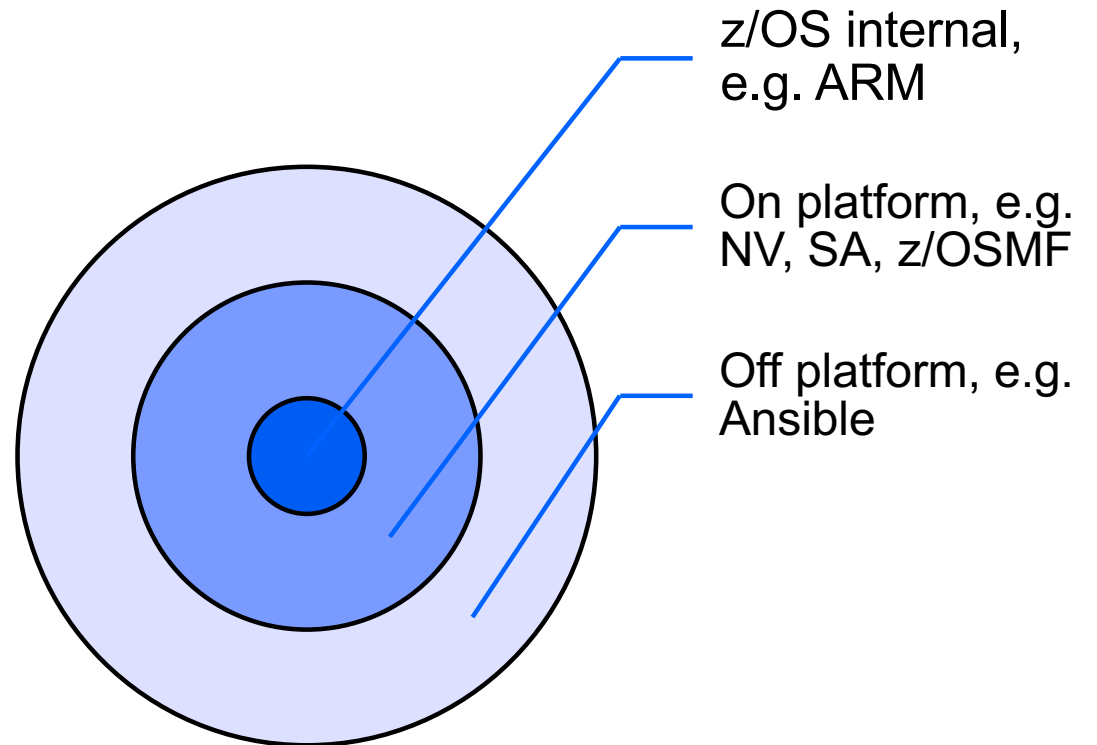
Principle: Automate as close to the source as possible

Faster

Less dependencies

More robust

Delegation of responsibilities



UrbanCode Deploy for Z & Ansible for Z – Better Together

If I have Ansible, how can UrbanCode improve my DevOps experience on z/OS?

Orchestrate the deployment of multiple packages simultaneously

Manage external events inside the execution of a deployment (e.g., approvals)

Restore a previous version in case of aborted deployment

Keep track of every delivery through inventory capabilities

Make deployment process easy to understand

If I have UrbanCode, how can Ansible improve my DevOps experience on z/OS?

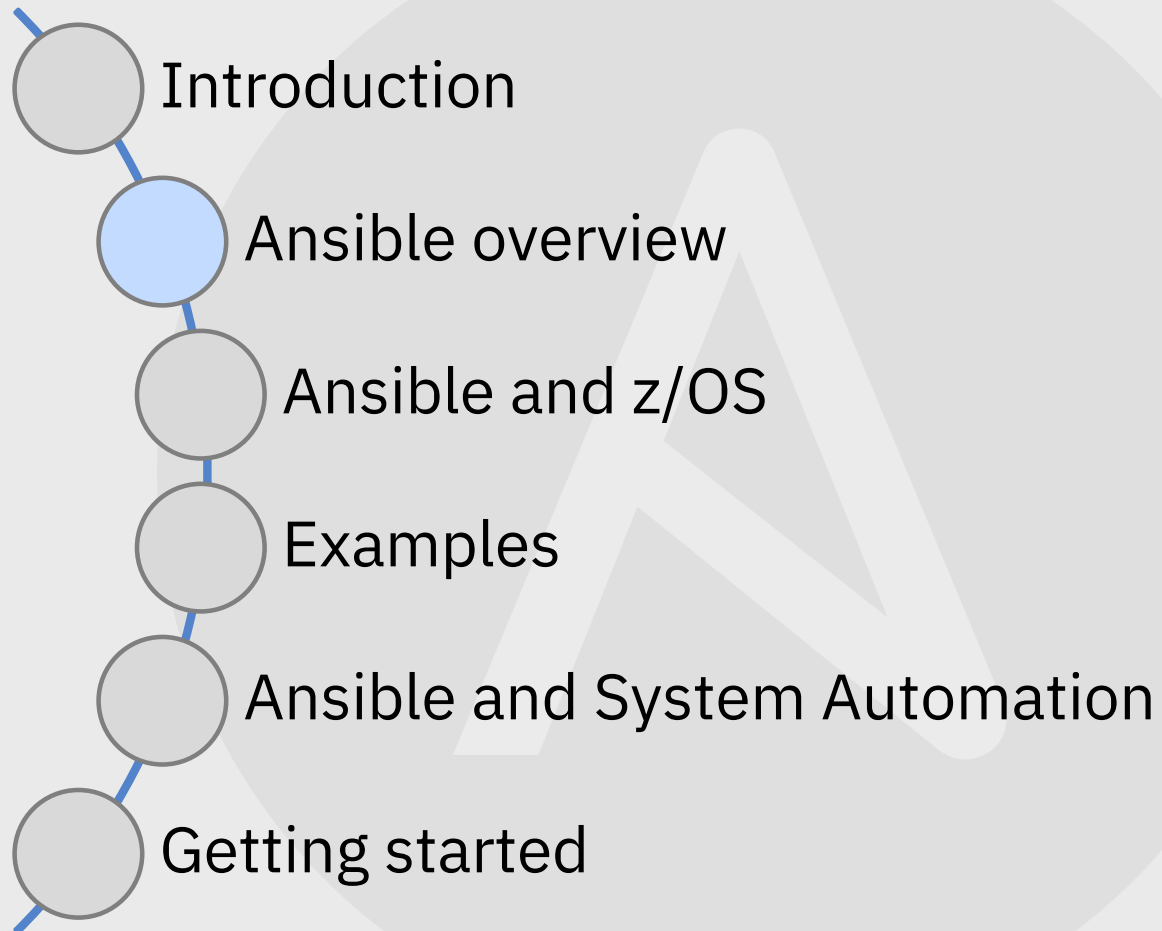
Limitless adaptation to user needs and integration with open-source tools

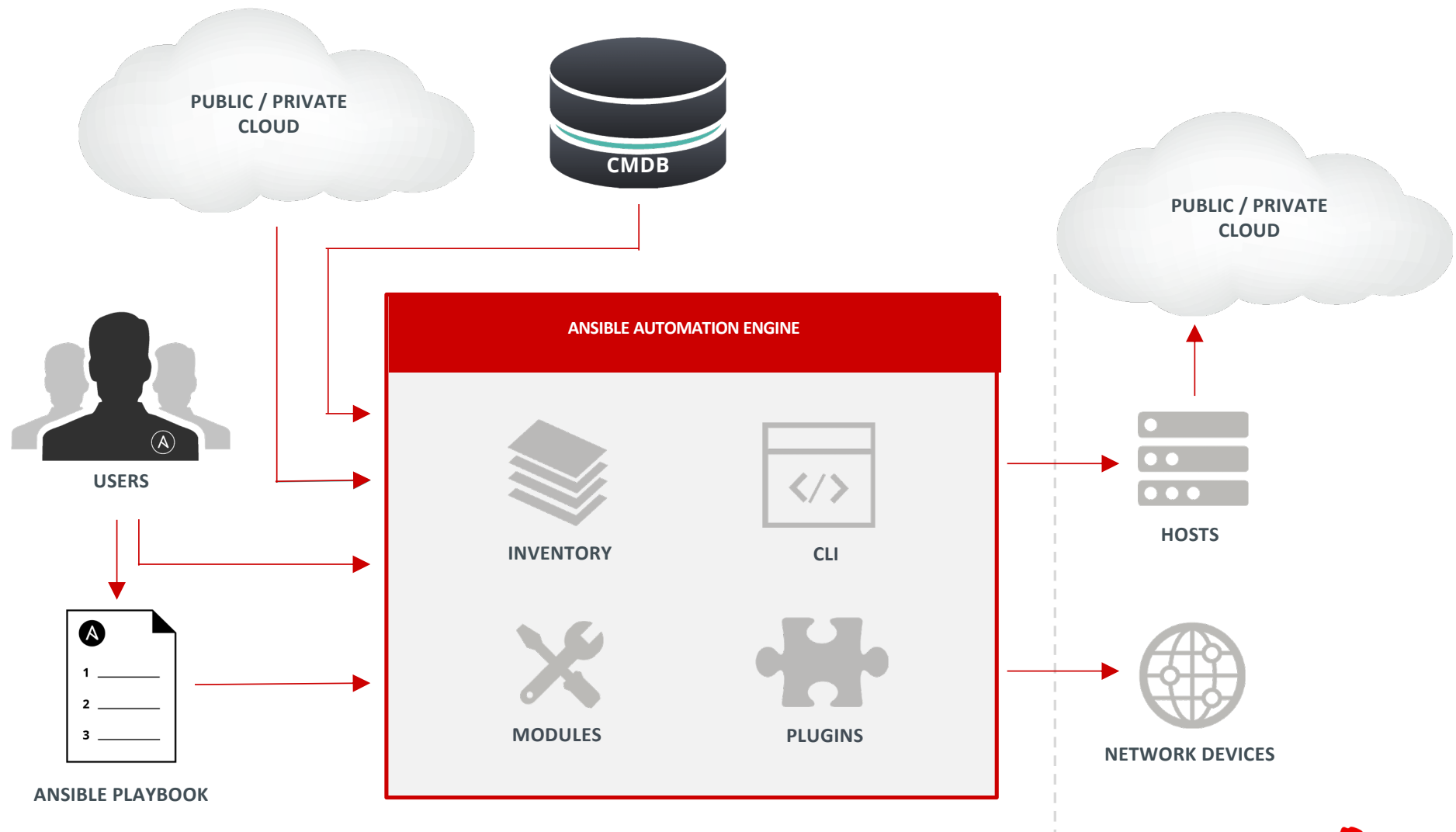
Keep advantage of agentless solution

Parallelism –ability to drive automation to multiple systems simultaneously

Consistency across enterprise hybrid multi-cloud environments through a single coherent market leading solution

Agenda





Ansible Control Nodes

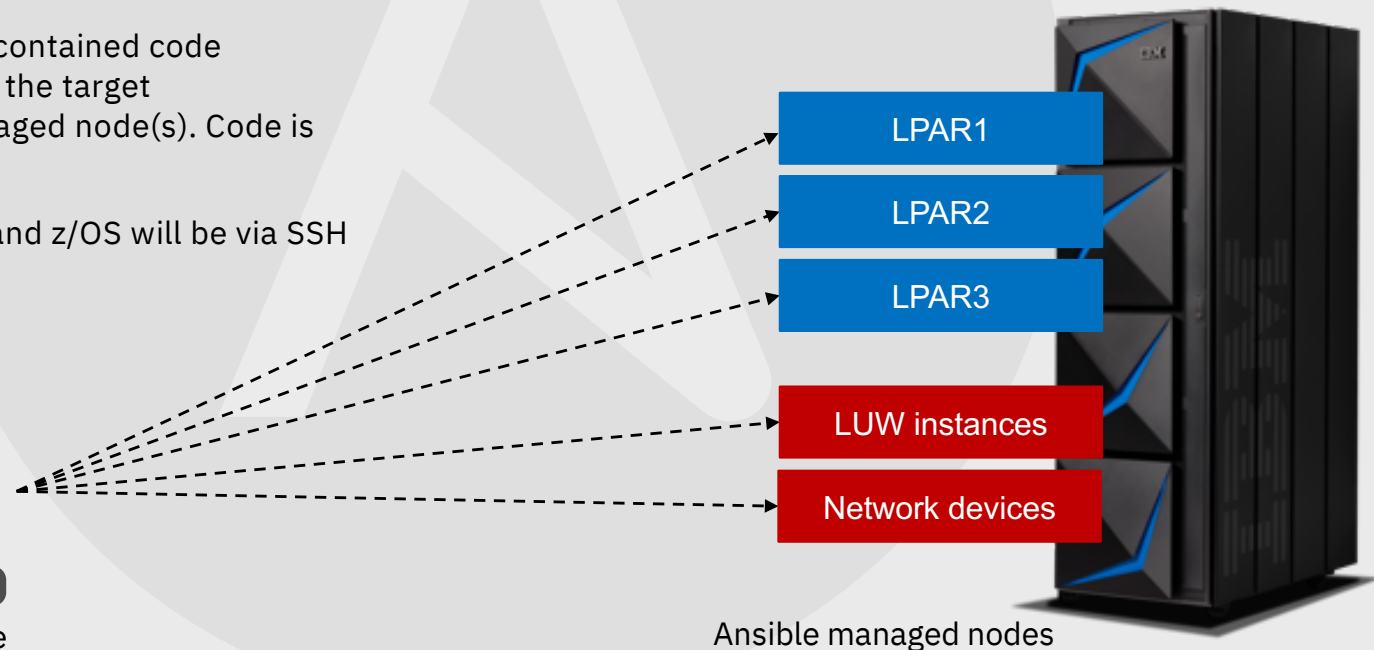
- The orchestrator
- Connects to 1-n managed nodes
- Playbook execution happens on the control node
 - invoke via Ansible Tower or Ansible AWX server
 - invoke Ansible CLI
- Most tasks will push small self-contained code (typically Python) to execute on the target environment(s) - aka – the managed node(s). Code is removed after execution.
- Connections to IBM Z systems and z/OS will be via SSH or via REST / SOAP APIs



Ansible control machine

Ansible Managed Nodes

- Some examples of managed nodes for z/OS could be:
 - Native z/OS LPARs
 - Linux on Z LPARs
 - z/VM guests
 - zD&T instances
 - KVM LPARs and guests



Ansible Inventory & Playbooks

Inventory

Groups

Setup variables

Targets

```
[all:vars]
ansible_connection=ssh # actually default mode smart is OK
ansible_port=22
ansible_user=ibmuser
ansible_python_interpreter=/usr/bin/python
TCPIP_PROFILE: USER.PARMLIB(TCPPROF2)

[zos]
10.149.60.189
```

Play Book

Group from Inventory

Setup variables

Set of Tasks

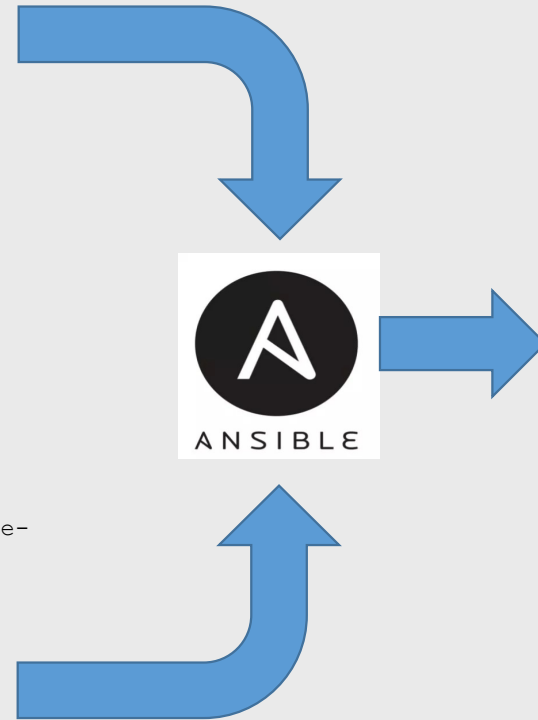
Modules

```
- hosts: zos
vars: ZCON_HTTP_PORT= '1322' ZCON_HTTPS_PORT= '1321'
tasks:
- name: Run create zOSConnectServer script
  shell: ./create-zOSConnectServer.j2
  args:
    chdir: '{{ uss_file_path }}'
    executable: /bin/sh

- name: Create connection config directory
  file:
    path: '{{ ZCEE_CONFIG_PATH }}/resources/imsmobile-
    config/connections'
    state: directory
    mode: '1777'

- name: Create new server proc
  zos_job_submit:
    name: '{{ uss_file_path }}/create-server-proc.j2'
    location: USS

- name: Copy rexx script for starting proc
  copy:
    src: '{{playbook_dir }}/files/scripts/start-region.rexx'
    dest: '{{ uss_utilities_path }}/start-region.rexx'
```



Ansible Playbook Execution

Execute playbook

```
$ ansible-playbook ansible/playbook.yml -i ansible/hosts.ini -l vb
```

Execution output

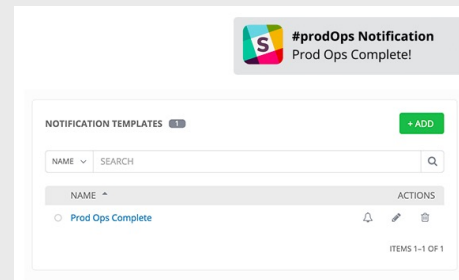
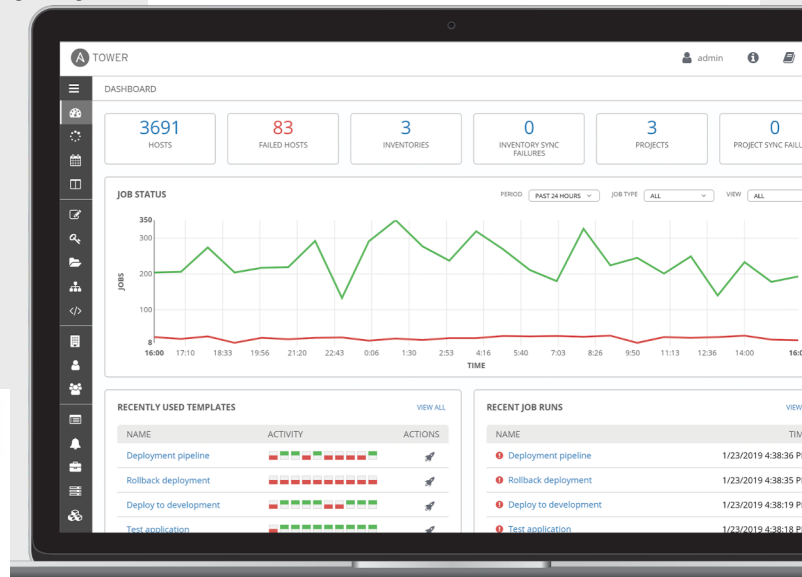
```
PLAY [all] *****
TASK [setup] *****
ok: [192.168.33.10]
TASK [ping] *****
ok: [192.168.33.10]
PLAY RECAP *****
192.168.33.10 : ok=2 changed=0 unreachable=0 failed=0
```

Playbooks are nice – how do we enterprise-ify them?

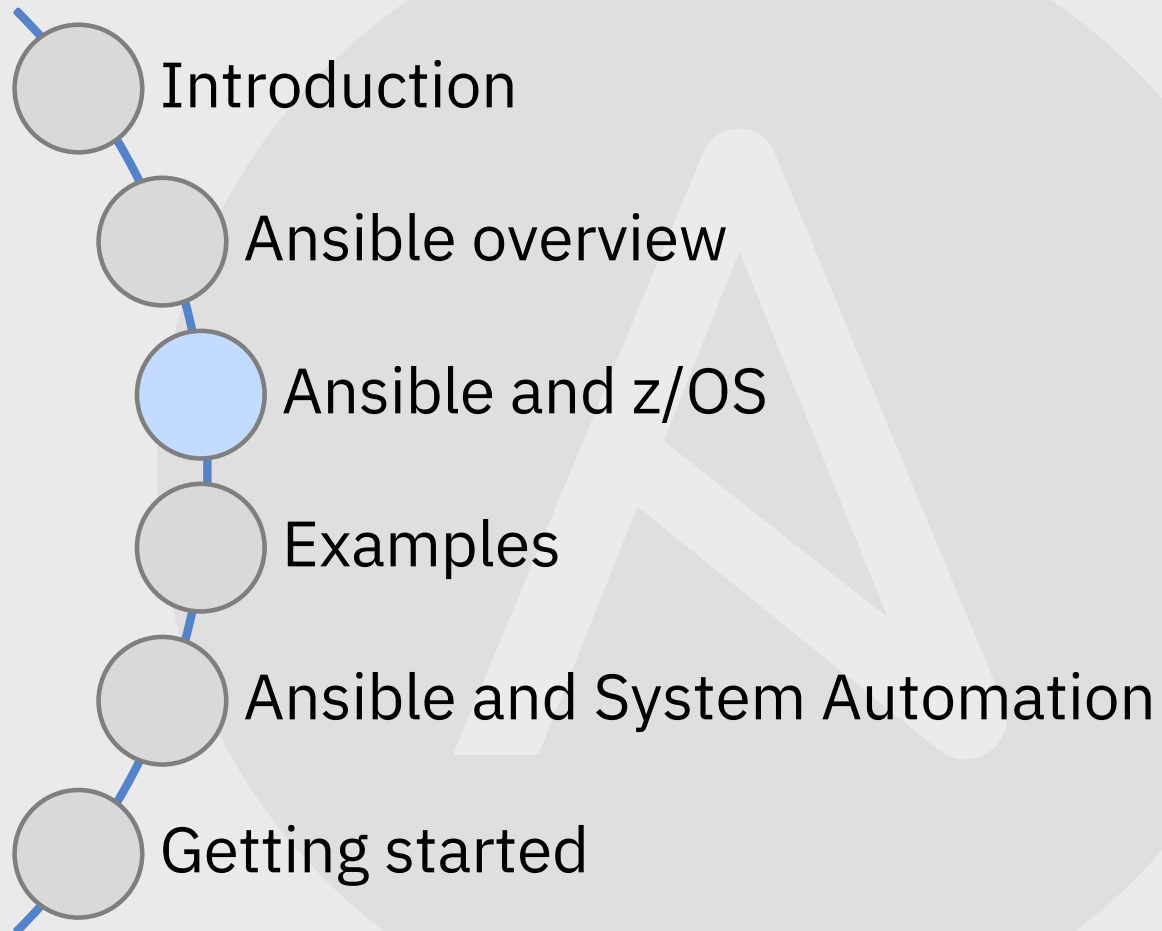
Playbooks on their own let individual execute tasks from command line but this doesn't scale well when you are talking about automating enterprise processes.

Enter **Ansible Tower** or its open source version **Ansible AWX**

- Centralize and control IT infrastructure – server that allows existing Ansible assets like playbooks to be imported and managed (via UIs, CLIs & APIs)
- Role-based access control - attach securely stored credentials to playbooks
- Job Templates - Playbook chaining to model complex processes
- Job scheduling, monitoring, audit trail
- Integrated Notifications
- Progress shown in real time
- Graphical inventory management




Agenda



z/OS & Ansible

- Ansible strategy for z/OS
 - Ansible is an extremely **flexible and extensible platform/framework** that allows clients to architect their user's experiences
 - Clients are telling us **Ansible is strategic for every other platform** and are exploring possibilities with z/OS
 - **Consolidate and normalize skills** for provisioning, deployment, orchestration in a single ecosystem wherever it makes sense
 - **Leverage Ansible as an orchestrator of orchestrators** across z/OS LPARs or across platforms
 - **Leverage existing automation assets** (scripts, REXX, JCL, z/OSMF, tools, etc), don't rewrite automation but use Ansible to help orchestrate and drive those assets in a consistent controlled way
 - **Simplify** configuration and deployment management
- What we are **NOT** advocating with Ansible
 - Ansible is not a replacement for z/OS operational automation provided by System Automation for z/OS or Netview that require **tight integration and fast reaction to events and messages**. However, as Ansible orchestrates provisioning of z/OS environments it must ensure these solutions are notified of Ansible activity so they can effectively monitor the products/environments that were provisioned.

Red Hat Ansible Certified Content for IBM Z: collections

 Available on
Automation Hub

Available on Ansible Galaxy [here](#)

IBM Z System Automation
Collection



IBM z/OS Core Collection



IBM z/OSMF Collection




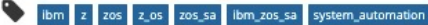















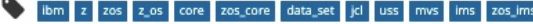





IBM z/OS CICS Collection



IBM z/OS IMS Collection



IBM Z Hardware Management
Console Collection

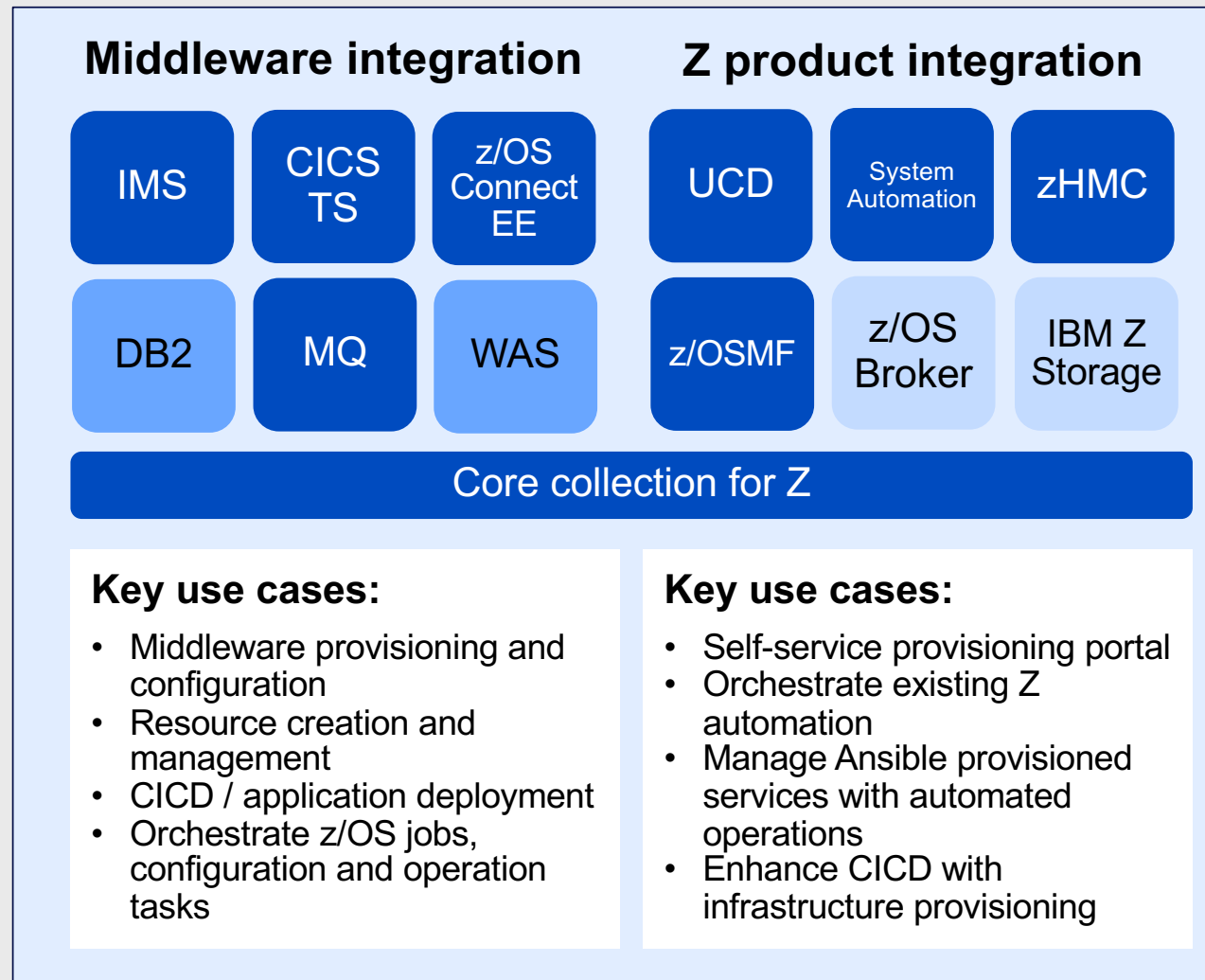
 ibm	ibm_zos_sysauto The IBM Z System Automation collection includes roles and sample playbooks to access the IBM Z System Automation Operations REST server. 	 5 / 5 Score  154 Downloads Current Version: 1.0.0 uploaded 4 months ago
 ibm	ibm_zos_core The IBM z/OS core collection includes connection plugins, action plugins, modules, filters and ansible-doc to automate tasks on z/OS. 	 5 / 5 Score  35033 Downloads Current Version: 1.4.0-beta.1 uploaded 24 days ago
 ibm	ibm_zosmf Ansible collection consisting of modules and roles to work with z/OS based on z/OS Management Facility(z/OSMF). 	 5 / 5 Score  17 Downloads Current Version: 1.0.1 uploaded 3 months ago
 ibm	ibm_zos_cics The Red Hat Ansible Certified Content for IBM Z CICS collection includes connection plugins, action plugins, modules and sample playbooks to automate tasks for CICS 	 5 / 5 Score  634 Downloads Current Version: 1.0.3 uploaded a month ago
 ibm	ibm_zos_ims The IBM z/OS IMS collection includes modules and sample playbooks to automate tasks for IBM IMS. 	 5 / 5 Score  9648 Downloads Current Version: 1.1.0 uploaded 6 months ago
 ibm	ibm_zhmc This collection can manage platform resources on IBM Z and LinuxONE machines, for example partitions, adapters, the Z system itself, or various resources on its Hardware Management Console (HMC). 	 1282 Downloads Current Version: 0.10.0 uploaded 2 months ago

Ansible ecosystem across IBM Z portfolio

Build up an Ansible ecosystem for IBM Z products

Deliver client needs through unified and powerful automation across products and platforms with Ansible

NOTE: Use-cases that can be achieved with Ansible and IBM Z are not limited to the provided integrations, automate today with APIs, commands, jobs, and more.

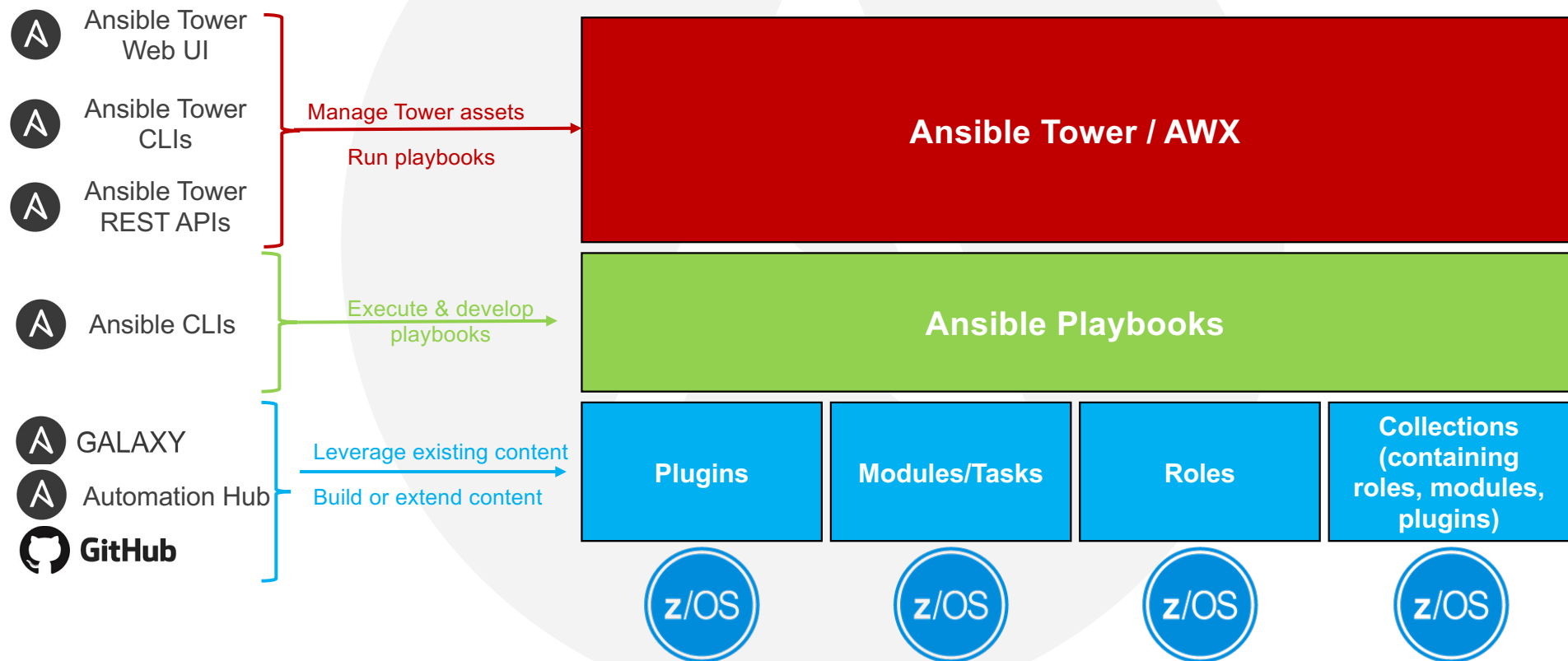


Ansible content available today

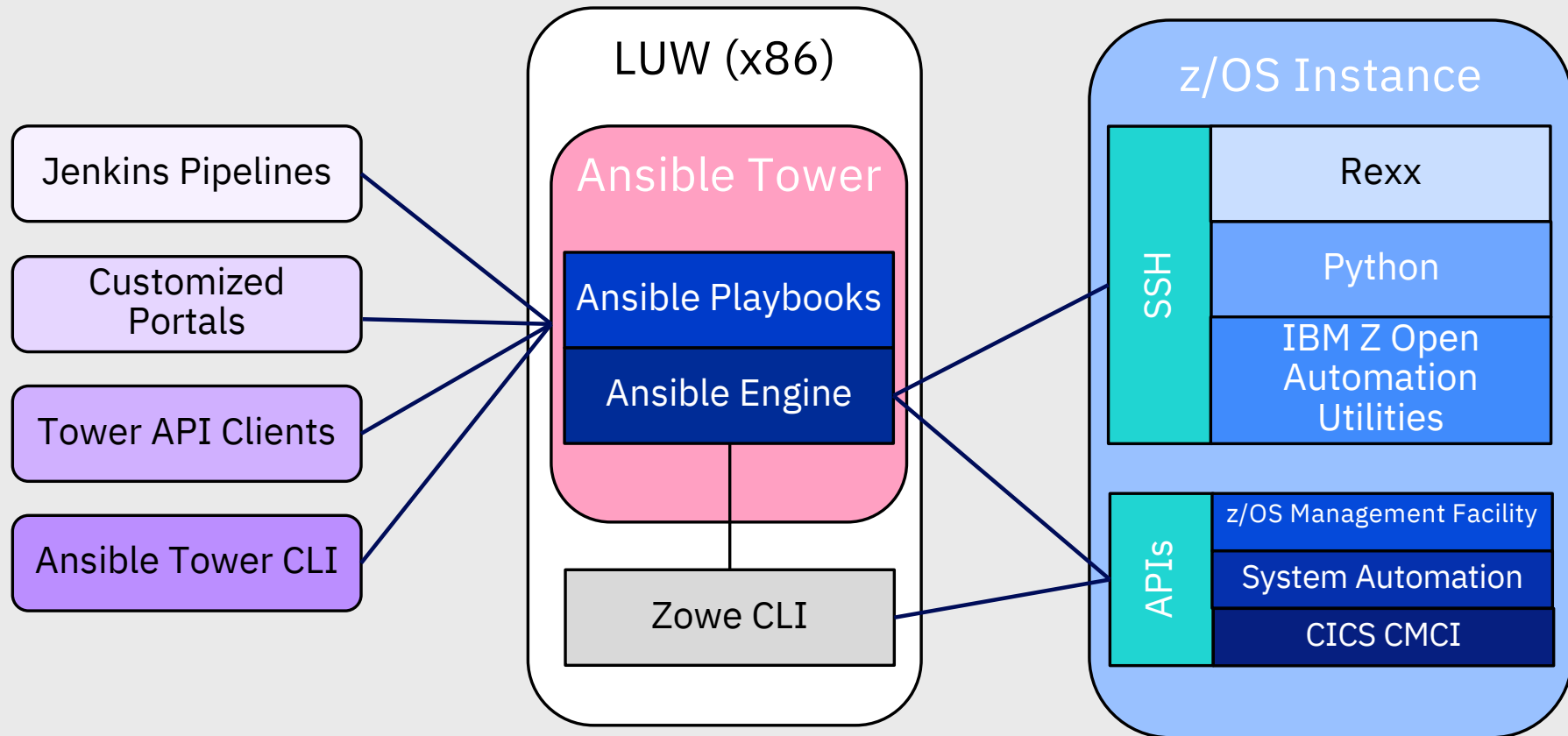
Collection / Samples coming soon

Content for consideration in future

Ansible Automation Platform and z/OS



Ansible in the broader landscape...



Ansible Collections for z/OS

- IBM provides Ansible Collections for the z/OS ecosystem
- Ansible Collection – z/OS core – `ibm.ibm_zos_core`
 - Provide easy to use **z/OS building blocks** necessary for customers to write automated playbooks and roles for z/OS
 - Building blocks may include:
 - Job related tasks ([submit jobs](#), [query jobs](#), [retrieve job output](#))
 - Data set tasks ([allocate](#), [delete](#), [copy](#), [rename](#), etc)
 - Issuing commands ([operator](#), [TSO](#), [ISPF commands](#))
 - **Certified Collection** available now
- Ansible Collections – z/OS middleware
 - Create Ansible Collections for middleware starting with IMS, CICS for interacting with & managing middleware components
 - Provide modules that wrap the key utilities provided by each middleware product necessary for management and maintenance
 - Roles will provide reusable procedures (utilizing multiple modules in sequence) to perform common middleware tasks (database reorganization, resource deployment, etc)

z/OS & Ansible

- **Batteries Included:** Out of the box Ansible can communicate & automate z/OS
 - Secure Shell (SSH) into z/OS Unix System Services (USS) to
 - Execute USS commands, scripts, submit JCL, copy data to/from z/OS
 - Invoke RESTful/SOAP APIs
 - Many products on z/OS support RESTful/SOAP APIs
 - Ansible has existing modules, such as the 'uri' module that can make calls to these APIs

Ansible for z/OS

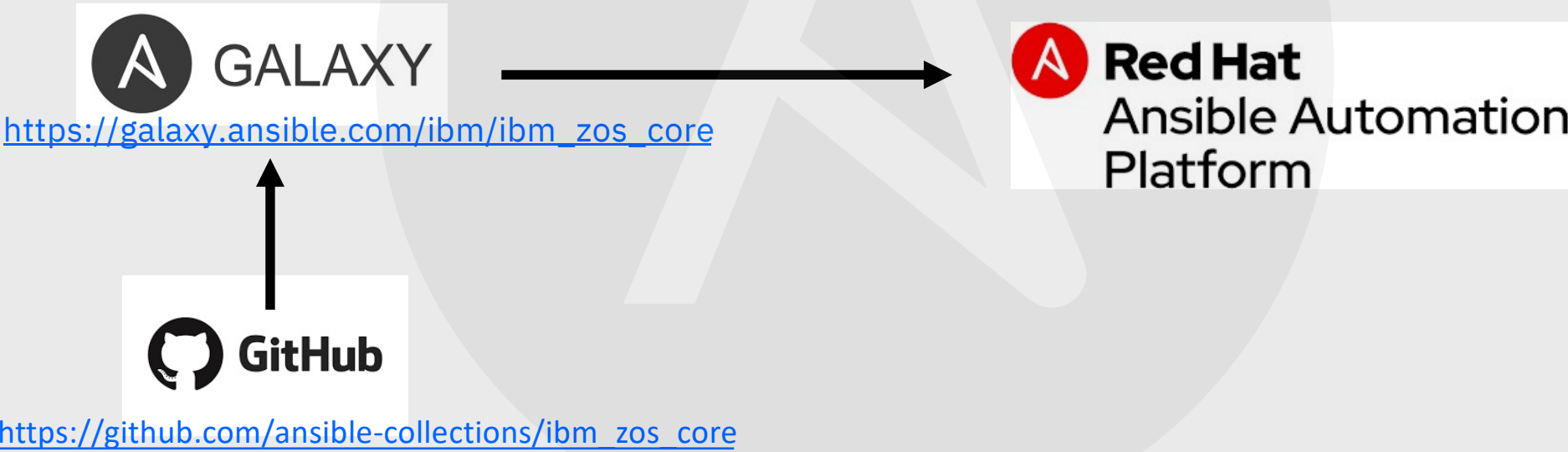
Modules that work out of the box

Ansible Module	Description
copy	Copy files: <ul style="list-style-type: none">• from control node to managed node (upload)• on the managed node (local copy) Works for z/OS Unix System Services files
fetch	Fetch files: <ul style="list-style-type: none">• from managed node to the control node Works for z/OS Unix System Services files
file	create and delete files/directories/links and set attributes/permissions Works for z/OS Unix System Services files
uri	Invoke REST/SOAP APIs z/OS products that are API enabled can be invoked using the uri module
shell	Executes a command on the managed node Works for z/OS Unix System Services files
command	See 'shell' module

Red Hat Ansible Certified Content for IBM Z

Develop content collections in the open on GitHub and Ansible Galaxy communities

Certified content collections supported by Red Hat (and IBM) with subscription to Ansible Automation Platform



Red Hat Ansible Content for IBM Z - Ansible Galaxy



Explore the [ibm_zos_core](#) collection

Key Collection Links



ibm

ibm_zos_core

The IBM z/OS core collection includes connection plugins, action plugins, modules, filters, sample playbooks and ansible-doc to automate tasks on z/OS.

5 / 5 Score 4788 Downloads

[Unfollow Collection](#)

[Issue Tracker](#)

[Repo](#)

[Website](#)

[Docs Site](#)

Details Read Me **Content**

Filter content...

Show: Roles Modules Playbooks Plugins

zos_data_set

Manage data sets

Module

zos_encode

Perform encoding operations.

Module

zos_job_output

Display job output

Module

zos_operator_action_query

Display messages requiring action

Module

zos_mvs_raw

Run a z/OS program.

Module

zos_lineinfile

Manage textual data on z/OS

Module

zos_copy

Copy data to z/OS

Module

zos_tso_command

Execute TSO commands

Module

zos_job_query

Query job status

Module

zos_job_submit

Submit JCL

Module

zos_operator

Execute operator command

Module

zos_fetch

Fetch data from z/OS

Module

Red Hat Ansible Content for IBM Z - Ansible Galaxy



Explore the [ibm_zos_ims](#) collection



ibm

ibm_zos_ims

The IBM z/OS IMS collection includes modules and sample playbooks to automate tasks for IBM IMS.

Key Collection Links

5 / 5 Score 54 Downloads

Follow Collection

Issue Tracker

Repo

Website

Docs Site

Details

Read Me

Content

Filter content...

Show: Roles Modules Playbooks Plugins

ims_catalog_populate

Add records to the IMS Catalog

Module

ims_acb_gen

Generate IMS ACB

Module

ims_psb_gen

Generate IMS PSB

Module

ims_dbrc

Submit IMS DBRC Commands

Module

ims_command

Submit IMS Commands

Module

ims_dbd_gen

Generate IMS DBD

Module

ims_catalog_purge

Purge records from the IMS Catalog

Module

catalog

Type: module_utils

Plugin

Red Hat Ansible Content for IBM Z - Ansible Galaxy




Explore the [ibm_zos_sysauto](#) collection

The screenshot shows the Ansible Galaxy interface for the **ibm_zos_sysauto** collection. At the top left is the IBM logo. The collection name **ibm_zos_sysauto** is displayed with a brief description: "The IBM Z System Automation collection includes roles and sample playbooks to access the IBM Z System Automation Operations REST server." To the right, a bracket groups several key collection links: "5 / 5 Score", "5 Downloads", "Login to Follow", "Issue Tracker", "Repo", "Website", and "Docs Site". Below the header are navigation tabs for "Details", "Read Me", and "Content". A filter section allows users to show "Roles", "Modules", "Playbooks", and "Plugins", all of which are currently selected. Two role cards are visible: **sa_create_dynamic_resource** (Role) with the description "Create and resume a new dynamic resource instance from a template in an IBM Z System Automation envi...", and **sa_delete_dynamic_resource** (Role) with the description "Delete a dynamic resource instance from the list of automated instances in IBM Z System Automation e..."

Red Hat Ansible Content for IBM Z - Ansible Galaxy



Explore the [ibm_zhmc](#) collection



ibm_zhmc

This collection can manage platform resources on IBM Z and LinuxONE machines, for example partitions, adapters, the Z system itself, or various resources on its Hardware Management Console (HMC).

2 Downloads

[Login to Follow](#) [Issue Tracker](#) [Repo](#)

[Website](#) [Docs Site](#)

Details **Read Me** **Content**

Filter content... Show: Roles Modules Playbooks Plugins

zhmc_crypto_attachment Module Attach crypto resources to partitions	zhmc_cpc Module Update CPCs	zhmc_storage_group Module Create storage groups
zhmc_storage_volume Module Create storage volumes	zhmc_adapter Module Update adapters and create Hipersocket adapters	zhmc_partition Module Create partitions
zhmc_user Module Create HMC users	zhmc_hba Module Create HBAs in partitions	zhmc_storage_group_attachment Module Attach storage groups to partitions
zhmc_virtual_function Module Create virtual functions in partitions	zhmc_nic Module Create NICs in partitions	common Plugin Type: <code>module_utils</code>

Key Collection Links

Red Hat Ansible Content for IBM Z - Ansible Galaxy



Explore the [ibm_zos_cics](#) collection



ibm

ibm_zos_cics

The Red Hat Ansible Certified Content for IBM Z CICS collection includes connection plugins, action plugins, modules and sample playbooks to automate tasks for CICS

Key Collection Links

5 / 5 Score 160 Downloads

[Follow Collection](#)

[Issue Tracker](#)

[Repo](#)

[Website](#)

[Docs Site](#)

Details

Read Me

Content

Filter content...

Show: Roles Modules Playbooks Plugins

cmci_action

Module

Perform actions on CICS and CICSplex SM resources

cmci_create

Module

Create CICS and CICSplex SM definitions

cmci_delete

Module

Delete CICS and CICSplex SM resources

cmci_get

Module

Query CICS and CICSplex SM resources and definitions

cmci_update

Module

Update CICS and CICSplex resources and definitions

cmci

Plugin

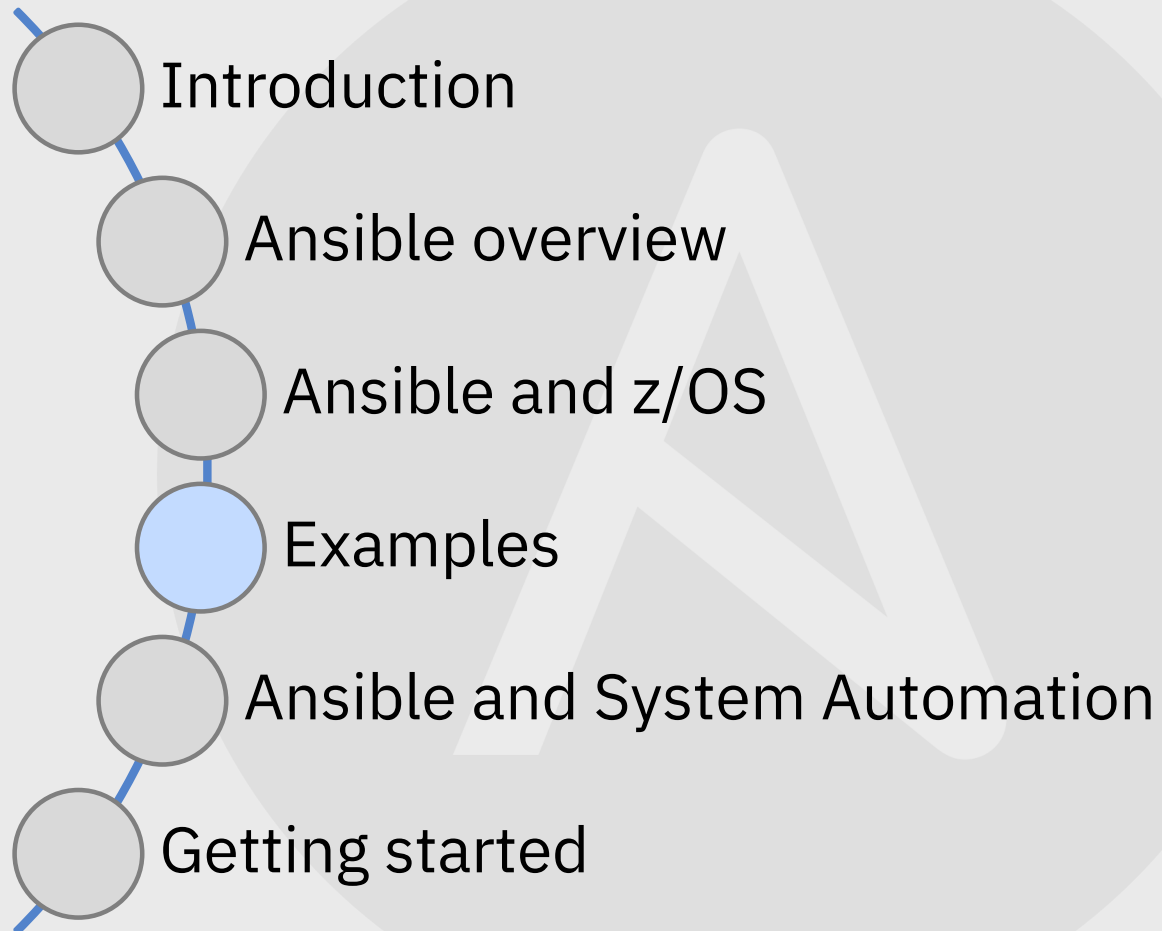
Type: `module_utils`

cmci

Plugin

Type: `doc_fragments_plugin`

Agenda



Examples & Demos

shell module & command module example

```
- name: Assemble DBD via command
  command: as -mOBJECT -I IMSBLD.I15RTSMM.SDFSMAc -I SYS1.MACLIB -o /tmp/DEDBJN21.o
  register: assemble_result
  shell: ld -o "'IMSTESTG.UDB.DBD.SOURCE(DEDBJN21)'" /tmp/DEDBJN21.o

- name: Print assembly return code {{ assemble_result.rc }}
  debug:
    msg:
      - "Assembly return code {{ assemble_result.rc }}"

- name: Link-Edit DBD via command
  shell: ld -o "'IMSTESTL.IMS1.DBDLIB(DEDBJN21)'" /tmp/DEDBJN21.o
  when: assemble_result.rc == 0
  register: linkedit_result

- name: Print link-edit return code {{ linkedit_result.rc }}
  debug:
    msg:
      - "Link-edit return code {{ linkedit_result.rc }}"
```

Examples & Demos

shell module & command module output:

```
TASK [Assemble DBD via command] *****
Sunday 23 February 2020 10:44:22 -0800 (0:00:00.040)      0:00:00.040 *****
changed: [zvm]

TASK [Print assembly return code 0] *****
Sunday 23 February 2020 10:44:29 -0800 (0:00:07.173)      0:00:07.214 *****
ok: [zvm] => {
  "msg": [
    "Assembly return code 0"
  ]
}

TASK [Link-Edit DBD via command] *****
Sunday 23 February 2020 10:44:29 -0800 (0:00:00.038)      0:00:07.252 *****
changed: [zvm]

TASK [Print link-edit return code 0]
*****
Sunday 23 February 2020 10:44:32 -0800 (0:00:02.757)      0:00:10.010 *****
ok: [zvm] => {
  "msg": [
    "Link-edit return code 0"
  ]
}
```

Examples & Demos

uri module example –
execute z/OS Management Facility requests via REST APIs:

```
- name: List jobs using z/OS Management Facility REST APIs
  uri:
    url: https://{{ ZOSMF_HOST }}:{{ ZOSMF_PORT
  }}/zosmf/restjobs/jobs?owner=*&prefix=ims1ctl
    user: "{{ ZOSMF_USER }}"
    password: "{{ ZOSMF_PASS }}"
    method: GET
    headers:
      Content-Type: application/json
    force_basic_auth: yes
    validate_certs: no
    status_code: 200
    return_content: yes
```

Examples & Demos

uri module output –

```
TASK [List jobs using z/OS Management Facility REST APIs] *****
Wednesday 26 February 2020 10:11:08 -0600 (0:00:00.031) 0:00:00.031 ****
ok: [tivlp02]

TASK [Print zosmf output] *****
Wednesday 26 February 2020 10:11:11 -0600 (0:00:02.576) 0:00:02.608 ****
ok: [tivlp02] => {
  "zosmf_result": {
    ...
    "json": [
      {
        "class": "STC",
        "files-url":
"https://tivlp02.svl.ibm.com:443/zosmf/restjobs/jobs/S0006064TIVLP02.D78665D9.....%3A/files",
        "job-correlator": "S0006064TIVLP02.D78665D9.....:",
        "jobid": "STC06064",
        "jobname": "IMS1CTL",
        "owner": "IBMUSER",
        "phase": 14,
        "phase-name": "Job is actively executing",
        "retcode": null,
        "status": "ACTIVE",
        "subsystem": "JES2",
        "type": "STC",
        "url":
"https://tivlp02.svl.ibm.com:443/zosmf/restjobs/jobs/S0006064TIVLP02.D78665D9.....%3A"
      }
    ],
    "status": 200,
    ...
  }
}
```

Examples & Demos

zos_data_set module example – **allocate a data set**, delete/replace if it already exists:

```
- name: allocate a data set
  zos_data_set:
    name: IMSTESTL.IMS2.DBDLIB
    type: PDS
    size: 4CYL
    format: U
    data class: SMS10
    replace: yes
```

```
TASK [allocate a data set] *****
Saturday 22 February 2020 14:10:17 -0800 (0:00:07.032)      0:00:07.071 *****
changed: [zvm]
```

Examples & Demos

zos_job_submit module examples:

```
- name: submit a job
  zos_job_submit:
    src: USER.PRIVATE.PROCLIB(DBDGEN)
    location: DATA_SET
    wait: true
    return_output: true
    register: job_detail
```

Output:

```
TASK [submit a job] *****
Saturday 22 February 2020 14:22:58 -0800 (0:00:00.030) 0:00:00.030 *****
changed: [zvm]
```

Examples & Demos

zos_job_query module examples

```
- name: Job query "{{ job_detail.jobs[0].job_id }}"
  zos_job_query:
    job_id: "{{ job_detail.jobs[0].job_id }}"
    register: job_detail

- name: Print query job details
  debug:
    msg:
      - "Job return code bad: {{ job_detail.jobs[0].ret_code.code }}"
      - "{{ job_detail }}"
```

Examples & Demos

zos_job_query
output:

```
TASK [Job query "JOB00171"] *****
Saturday 22 February 2020 15:49:10 -0800 (0:00:00.058) 0:00:17.392 *****
ok: [zvm]

TASK [Print query job details] *****
Saturday 22 February 2020 15:49:14 -0800 (0:00:04.240) 0:00:21.632 *****
ok: [zvm] => {
  "msg": [
    "Job return code bad: 0000",
    {
      "changed": false,
      "failed": false,
      "jobs": [
        {
          "job_id": "JOB00171",
          "job_name": "DBDGEN00",
          "owner": "OMVSADM",
          "ret_code": {
            "code": "0000",
            "msg": "CC 0000"
          }
        }
      ]
    }
  ]
}
```


Examples & Demos

zos_job_output module examples

```
- name: Get job output "{{ job_detail.jobs[0].job_id }}"
  zos_job_output:
    job_id: "{{ job_detail.jobs[0].job_id }}"
    register: job_detail

- name: Print job output "{{ job_detail.jobs[0].job_id }}"
  debug:
    msg:
      - "{{ job_detail }}"
```

Examples & Demos

zos_job_output output:

```
TASK [Get job output "JOB00172"] *****
Saturday 22 February 2020 16:01:14 -0800 (0:00:00.043) 0:00:23.053 *****
ok: [zvm]

TASK [Print job output "JOB00172"] *****
Saturday 22 February 2020 16:01:18 -0800 (0:00:04.280) 0:00:27.333 *****
ok: [zvm] => {
  "msg": [
    {
      "changed": false,
      "failed": false,
      "jobs": [
        {
          "class": "K",
          "content_type": "JOB",
          "ddnames": [
            {
              "byte_count": "682",
              "content": [
                "1 JES2 JOB LOG -- SYSTEM STL1 -- NODE STL1",
                "0 ",
                "16.01.02 JOB00172 ---- SATURDAY, 22 FEB 2020 ----",
                "16.01.02 JOB00172 IRR010I USERID OMVSADM IS ASSIGNED TO THIS JOB.",
                "16.01.02 JOB00172 ICH70001I OMVSADM LAST ACCESS AT 16:00:59 ON SATURDAY, FEBRUARY 22, 2020",
                "16.01.02 JOB00172 $HASP373 DBDGEN00 STARTED - INIT 15 - CLASS K - SYS STL1",
                "16.01.03 JOB00172 SMF000I DBDGEN00 C ASMA90 0000",
                "16.01.03 JOB00172 SMF000I DBDGEN00 L IEWL 0000",
                "16.01.03 JOB00172 $HASP395 DBDGEN00 ENDED - RC=0000",
                "0----- JES2 JOB STATISTICS -----",
                "- 22 FEB 2020 JOB EXECUTION DATE",
                "- 28 CARDS READ",
                "- 158 SYSOUT PRINT RECORDS",
                "*****"
              ]
            }
          ]
        }
      ]
    }
  ]
}
```

Examples & Demos

zos_operator module example

```
- name: Issue a z/OS Operator/Console command
zos_operator:
  cmd: "D A,L"
  register: cmd_output

- name: Print command output
debug:
  msg:
    - "{{ cmd_output }}"
```

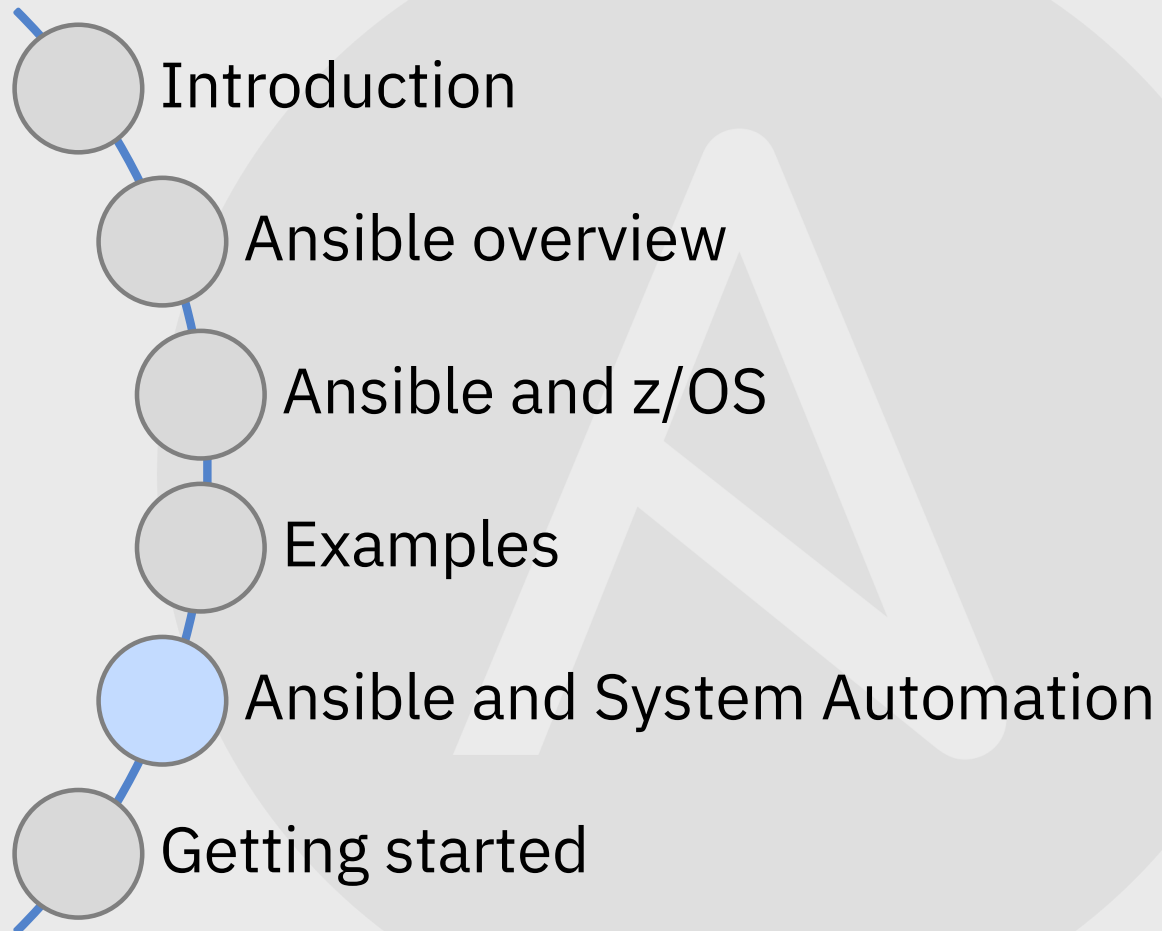
Examples & Demos

zos_operator
output:

```
TASK [Issue a z/OS Operator/Console command] *****
Sunday 23 February 2020 10:21:36 -0800 (0:00:00.030) 0:00:00.030 *****
changed: [zvm]

TASK [Print command output] *****
Sunday 23 February 2020 10:21:44 -0800 (0:00:07.847) 0:00:07.878 *****
ok: [zvm] => {
  "msg": [
    {
      "changed": true,
      "failed": false,
      "...":
      "rc": 0,
      "response": "EC01007 2020054 10:21:43.05 ISF031I CONSOLE OMVSADM ACTIVATED\n
EC01007 2020054 10:21:43.05 -D A,L\n
EC01007 2020054 10:21:43.06 IEE114I 10.21.43 2020.054 ACTIVITY 622\n
JOBS M/S TS USERS SYSAS INITS ACTIVE/MAX VTAM OAS\n
00018 00013 00001 00033 00035 00001/00020 00021\n
LLA LLA LLA NSW S VLF VLF VLF NSW S\n
JES2 JES2 IEFPROC NSW S RMF RMF IEFPROC NSW S\n
IGVDGNPP IGVDGNPP IGVDGNPP OWT S IMSVTAM IMSVTAM IEFPROC NSW S\n
TSO TSO STEP1 NSW S TCPIP TCPIP TCPIP NSW S\n
DHCPCLNT DHCPCLNT DHCPCLNT OWT S0 INETD INETD *OMVSEX OWT S\n
FTPD1 STEP1 OMVSKERN OWT A0 TN3270 TN3270 TN3270 NSW S\n
ICSF ICSF NSW S SSHD4 STEP1 OMVSKERN OWT A\n
AUTOSRVR JAVAJVM IN J0 ATRRRS ATRRRS RRS NSW S\n
IRLM1 IRLM1 IRLM NSW J SCI1 SCI1 IEFPROC NSW J\n
OM1 OM1 IEFPROC NSW J IMSCTL01 IMSCTL01 IEFPROC NSW J\n
IMSDLI01 IMSDLI01 IEFPROC NSW J IMSDBR01 IMSDBR01 IEFPROC NSW J\n
ODBM1 CSLDC001 IEFPROC NSW J ODBM2 CSLDC002 IEFPROC NSW J\n
OMVSADM STEP1 OMVSADM OWT A0 OMVSADM6 STEP1 OMVSADM OWT A\n
OMVSADM7 STEP1 OMVSADM TN A0 OMVSADM6 STEP1 OMVSADM TN A
```

Agenda



Hybrid automation joining the Red Hat Ansible Automation Platform with IBM Z System Automation

#9

of top ranked open source projects on GitHub make it a vibrant community

+3000 Modules

for automating many tasks in network, storage, server, applications and many more on different platforms, including IBM Z

3 Reasons

Ansible is already used by many IBM Z clients

Compelling choice for centralized automation and orchestration

DevOps teams use Ansible to quickly deploy dev and test environments already today

Cloud and DevOps lead to more dynamic compute environments on z/OS

Scenarios

The IBM z/OS Provisioning Toolkit, [zospt](#) is used to provision new instances of WebSphere Liberty Profiles within a few minutes

[IBM Cloud Provisioning and Management for z/OS](#) portal is used to provision unique instances of middleware (single or multiple address spaces) within minutes

[IBM z/OS Cloud Broker](#) leverages z/OS Cloud Provisioning and Management APIs to create a containerized application that include CICS, Db2, IMS, or MQ as a z/OS-based application tier

More and more clients use [Ansible](#) to deploy software on distributed platforms; with the availability of Ansible collections for z/OS, z/OSMF and IMS, they now also start to use it on z/OS

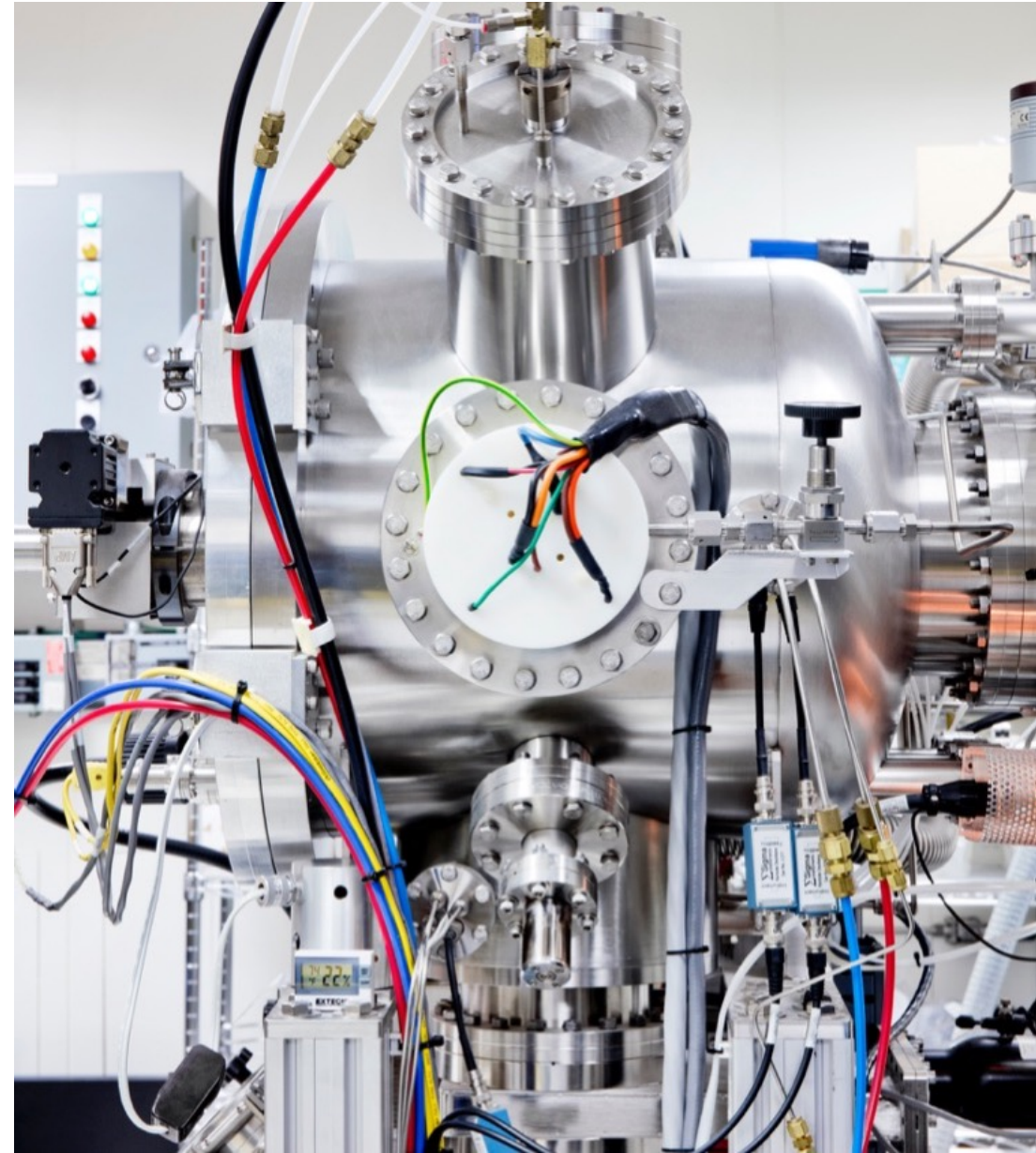
Challenges addressed

Issue

z/OS is already a highly automated system and products such as System Automation must constantly be updated when new workloads are provisioned to avoid that parts of the system cannot be managed anymore

Need

Ensure instant and automatic Visibility, Control, and Automation of any type of workload using existing management software, hence protecting customer investments



Remote access to z/OS automated operations through standard REST APIs

NV 6.3
OA59461

NetView and System Automation

Create automation tables / entries

Retrieve automation table statistics

Manage CANZLOG¹

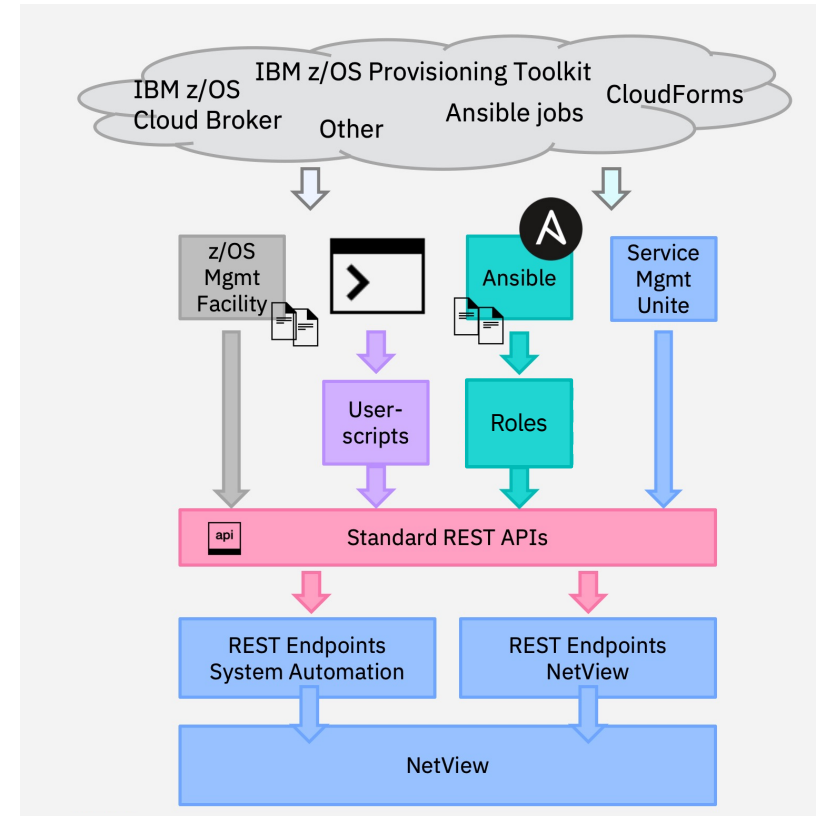
Retrieve DDVIPA² connection distribution statistics

Issue NetView commands

Create / delete dynamic resources

Manage resources (start, stop, suspend, resume)

1. CANZLOG = Consolidated Audit, NetView, and z/OS Log
2. DDVIPA = Distributed dynamic virtual IP address



Embedded API browser

Query templates

Manage resources

Templates Service to access templates that are defined in the current active IBM Z System Automation policy. These templates can be used to create so called dynamic resource instances. ✓

GET `/v1/templates` Find all templates defined in the loaded policy. 🔒

Resources Service to access resources currently managed by IBM Z System Automation. Resources are defined in the current active automation policy - and can be created dynamically following the definition in so called templates that are also defined in the current active automation policy. ✓

GET `/v1/resources` Find all managed resources. 🔒

POST `/v1/resources` Create a new dynamic resource. 🔒

GET `/v1/resources/{resourceId}` Get a managed resource by its ID. 🔒

DELETE `/v1/resources/{resourceId}` Delete a dynamically created resource by its ID. 🔒

POST `/v1/resources/{resourceId}/start` Send a start request for a managed resource. 🔒

POST `/v1/resources/{resourceId}/stop` Send a stop request for a managed resource. 🔒

POST `/v1/resources/{resourceId}/suspend` Send a suspend request for a managed resource. 🔒

POST `/v1/resources/{resourceId}/resume` Resume a managed resource. 🔒

CSE z/OS zExperten Forum / Vitznau (CH) / October 21, 2021 / © 2021 IBM Corporation

Join dynamic resources with automated processes

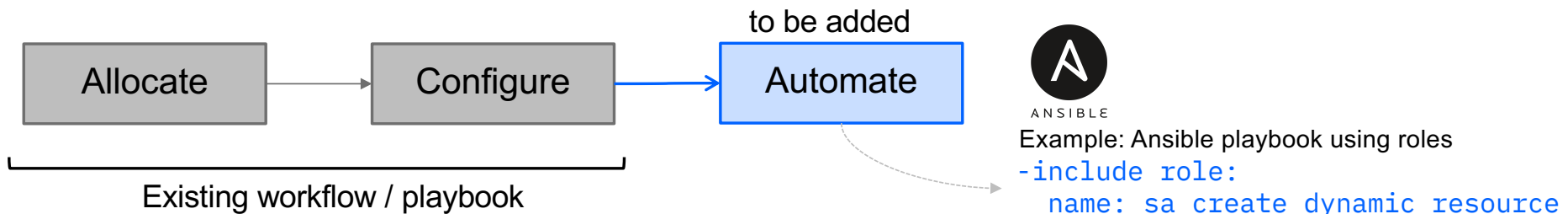
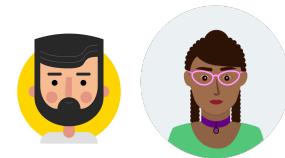
NV 6.3
OA59461

Existing automation capabilities for provisioning / de-provisioning

- z/OSMF workflows and tools such as the IBM z/OS Provisioning Toolkit
- IBM Cloud Provisioning and Management for z/OS
- Vendor-provided or homegrown scripts or batch solutions - or -
- **Ansible**



To let the **Operations team** manage such dynamic workloads, the corresponding dynamic resource definitions can be added to System Automation using REST services



Ansible integration with IBM Z System Automation

NV 6.3
OA59461



Executes
Playbook



ANSIBLE

Deb

New z/OS Developer

Ansible Playbook
Provision IMS Region
Create dynamic System
Automation resource and
start it

Step 1: Provision IMS region

- ✓ Allocate z/OS resources
- ✓ Create configuration

Step 2: Create System Automation resource

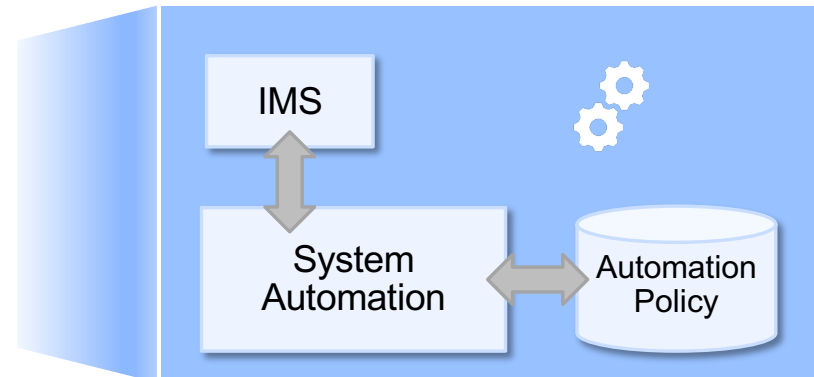
- ✓ Create dynamic resource from template in the automation policy
- ✓ Include resource into existing goal-oriented automation



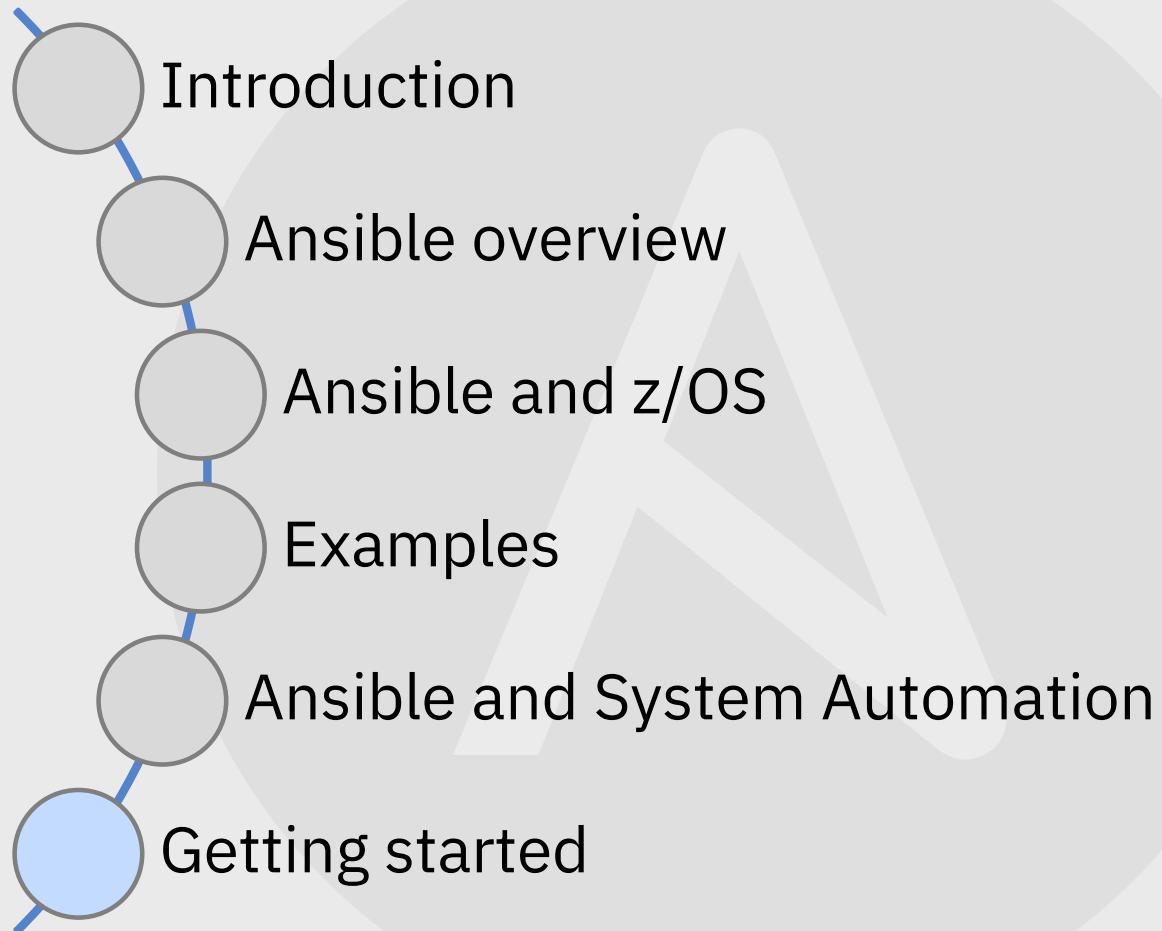
Operates z/OS
through System
Automation



Annette
Z IT Operator



Agenda



How to get started?

5 Simple Steps:

1. Enable OpenSSH
2. Install Python
3. Install IBM Z Open Automation Utilities (ZOA Utilities)
4. Install or update to Ansible 2.9+
5. Install the desired Ansible for z/OS collection from
 - Ansible Galaxy
 - Ansible Automation Hub (subscription required)

Pre-requisites (free with Optional S&S)

z/OS as a managed node

- Unix System Services (**USS**) enabled on z/OS
- Secure Shell (**SSH**) enabled on z/OS - [link](#)
- **Python 3.9.2** – IBM Open Enterprise SDK for Python
 - Download PAX archive - [link](#)
 - FMID HAMB390
 - SMP/E (CBPDO, ServerPac) (5655-PYT)
- IBM Z Open Automation Utilities (**ZOAU**) - [link](#)
 - FMID HAL5100

Ansible control node

- Ansible engine 2.9 or higher
- Ansible Tower 3.6+

Step 1: Try the Ansible® IBM Z Trial

Free guided virtual demo environment, providing users with the ability to run and review Ansible playbooks automating tasks on z/OS

[Register here](#)

IBM Z Trial

A Mainframe Automation with Red Hat Ansible

Welcome to your IBM Z Trial environment. Get started by exploring the scenarios below. Please approach the scenarios in order for an optimal experience.

SCENARIO | 10 MINS

Playbook for pingg
z/OS

Explore scenario

SCENARIO | 15 MINS

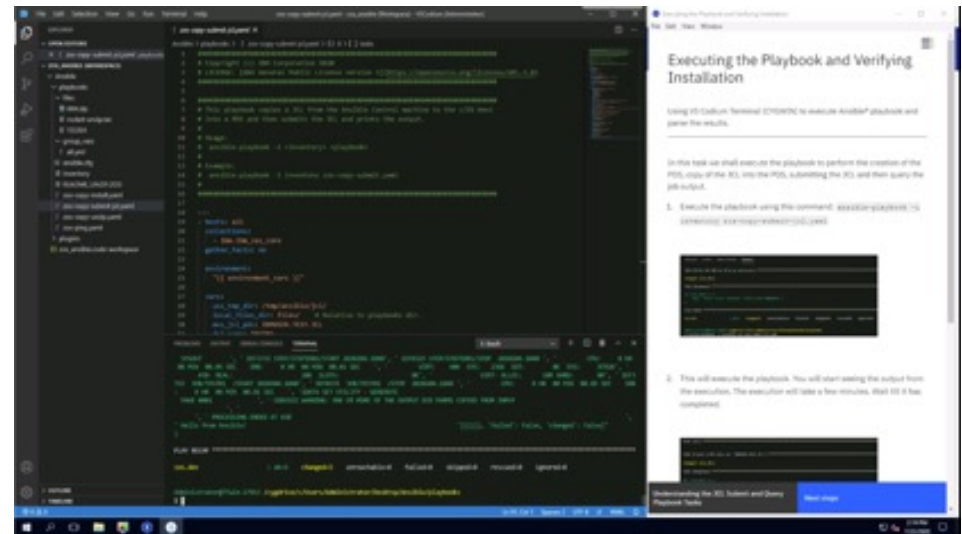
Playbook for z/OS
Copy and Install
Tarball onto USS

Explore scenario

SCENARIO | 15 MINS

Playbook for z/OS
Create PDS, Copy JCL,
Submit and Query Job

Explore scenario

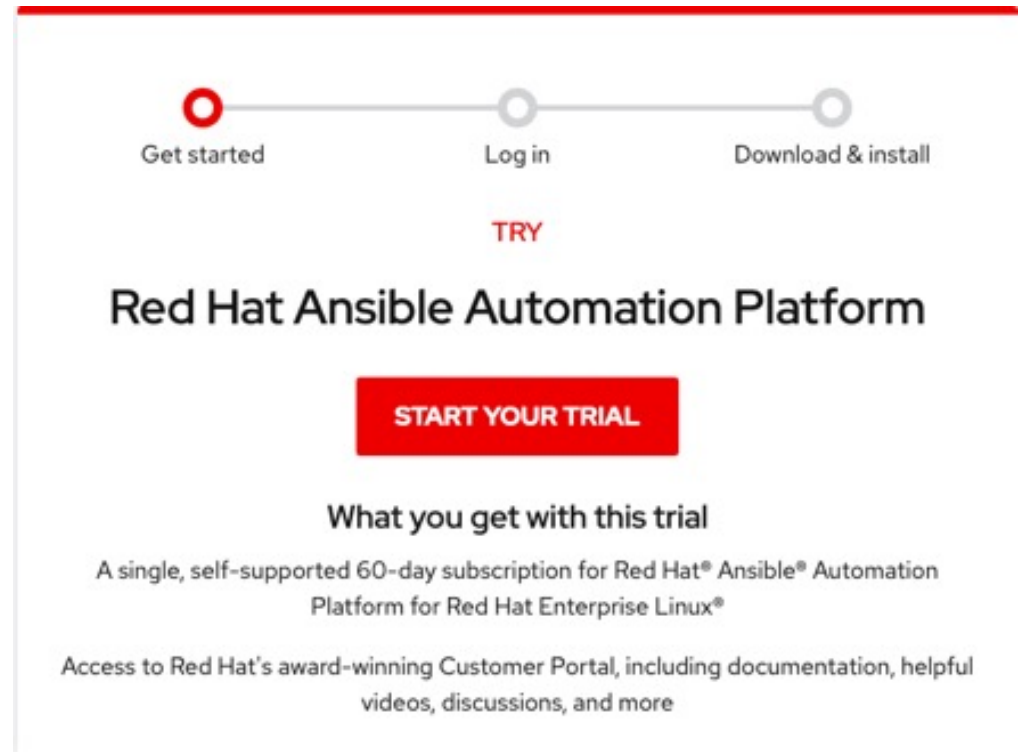


Step 2: Request the Ansible® Automation Platform free trial

Start your Ansible journey with the enterprise supported Ansible Automation Platform.

Includes:

- A single, self-supported 60-day subscription for Red Hat® Ansible® Automation Platform for Red Hat Enterprise Linux
- Access to Red Hat's award-winning Customer Portal, including documentation, helpful videos, discussions, and more
- Can support up to 100 Ansible managed nodes



The image shows a screenshot of the Red Hat Ansible Automation Platform trial registration page. At the top, there is a progress bar with three steps: 'Get started' (indicated by a red circle), 'Log in' (indicated by a grey circle), and 'Download & install' (indicated by a grey circle). Below the progress bar, the word 'TRY' is written in red. The main heading is 'Red Hat Ansible Automation Platform'. Below this, there is a prominent red button with the text 'START YOUR TRIAL'. Underneath the button, the text reads 'What you get with this trial'. This is followed by a description: 'A single, self-supported 60-day subscription for Red Hat® Ansible® Automation Platform for Red Hat Enterprise Linux®'. At the bottom, there is a line of text: 'Access to Red Hat's award-winning Customer Portal, including documentation, helpful videos, discussions, and more'.

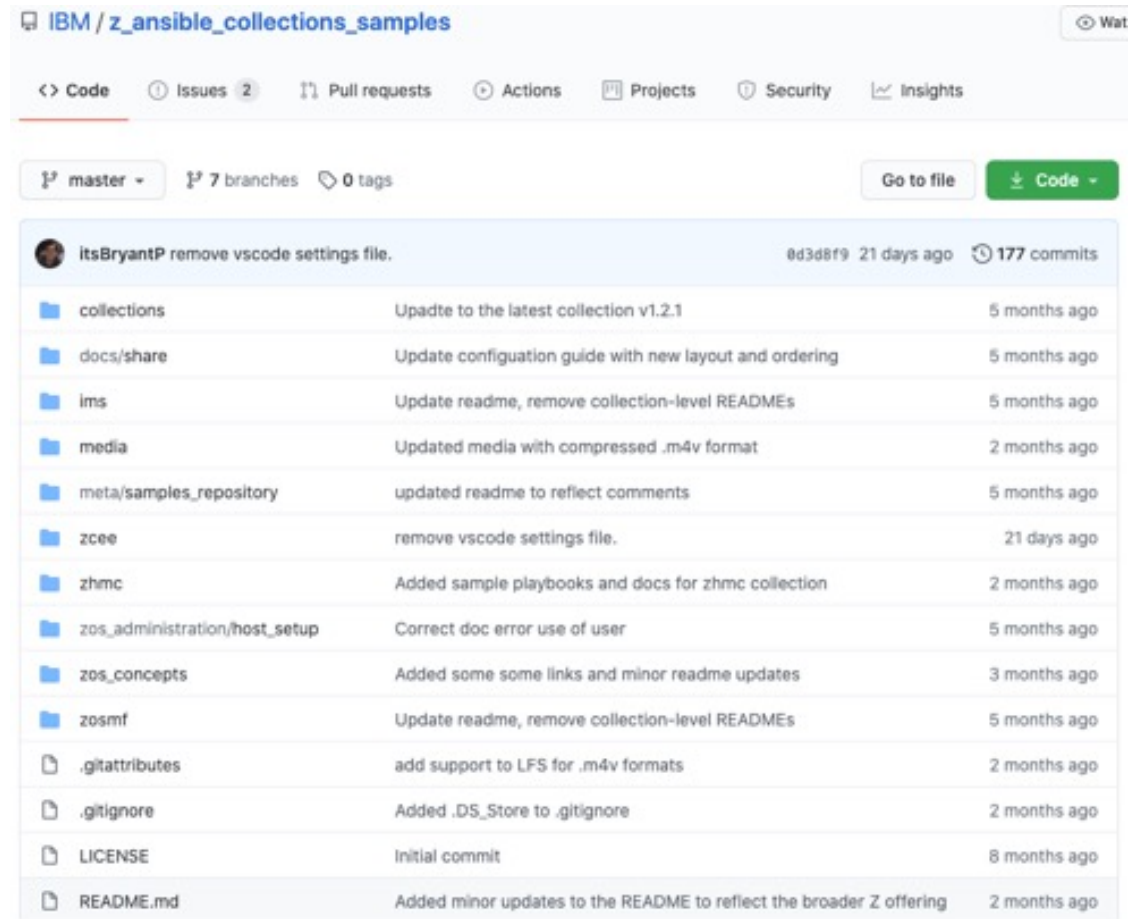
[Register here](#)

Step 3: Begin with the Ansible® for IBM Z sample playbook repository

Prewritten playbooks
spanning multiple use-cases
and technologies

Contribute your playbooks!

[Access here](#)



IBM / z_ansible_collections_samples

<> Code 1 Issues 2 Pull requests Actions Projects Security Insights

master 7 branches 0 tags Go to file Code

itsBryantP remove vscode settings file. 8d3d8f9 21 days ago 177 commits

collections	Upadte to the latest collection v1.2.1	5 months ago
docs/share	Update configuration guide with new layout and ordering	5 months ago
ims	Update readme, remove collection-level READMEs	5 months ago
media	Updated media with compressed .m4v format	2 months ago
meta/samples_repository	updated readme to reflect comments	5 months ago
zcee	remove vscode settings file.	21 days ago
zhmc	Added sample playbooks and docs for zhmc collection	2 months ago
zos_administration/host_setup	Correct doc error use of user	5 months ago
zos_concepts	Added some some links and minor readme updates	3 months ago
zosmf	Update readme, remove collection-level READMEs	5 months ago
.gitattributes	add support to LFS for .m4v formats	2 months ago
.gitignore	Added .DS_Store to .gitignore	2 months ago
LICENSE	Initial commit	8 months ago
README.md	Added minor updates to the README to reflect the broader Z offering	2 months ago

Join the community

- Facilitate communication amongst Guild members
- Host past recordings and presentations of Guild meetings
- Post blogs & videos for the IBM Z community



[Ansible for IBM Z:
IBM Z & LinuxONE
Community](#)

A screenshot of the website for the Ansible for IBM Z community. The page has a dark blue background with a pattern of binary code and cloud shapes. At the top, there is a navigation bar with the following items: "IBM Z and LinuxONE Community", "Get involved" (with a dropdown arrow), "Topic groups" (with a dropdown arrow), "User groups", "Solutions" (with a dropdown arrow), and "Resources". The main content area features the title "Ansible for IBM Z" in large white text, followed by the subtitle "Facilitate communication, user interaction and feedback for Red Hat Ansible Certified Content for IBM Z". Below this is a blue button labeled "Contribute" with a dropdown arrow. At the bottom, there is a dark navigation bar with the following links: "Home", "Blog entries", "Discussions", "Events", "Videos", and "Library".

IBM Z and LinuxONE Community Get involved ▾ Topic groups ▾ User groups Solutions ▾ Resources

Ansible for IBM Z

Facilitate communication, user interaction and feedback for Red Hat Ansible Certified Content for IBM Z

Contribute ▾

Home Blog entries Discussions Events Videos Library

New content in the community

- [Ansible for IBM Z collections](#)
- [Sample Playbook Repository](#)
- [Ansible for IBM Z community](#)
- [Ansible for IBM Z videos on IBM Media Center](#)
- [Triton Ansible blog series](#)
- [Ensono Terminal Talk Podcast](#)

Thank you

