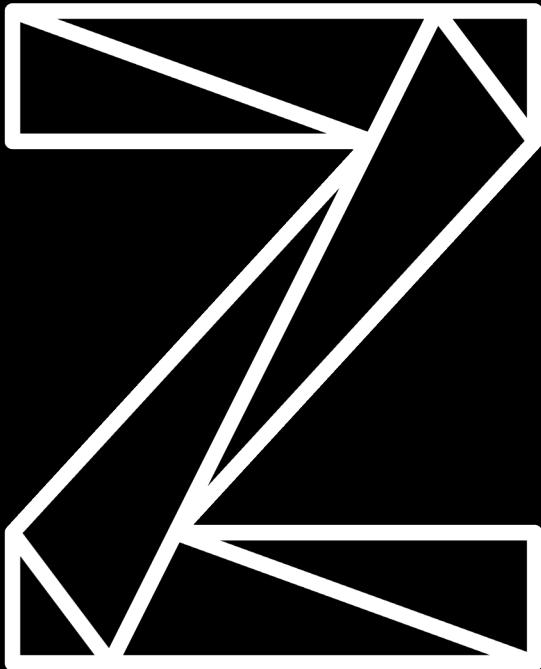


IBM CICS Transaction Server 5.6



Mark Cocker
CICS Transaction Server Product Manager
mark_cocker@uk.ibm.com



IBM CICS Transaction Server 5.6

CICS TS 5.6 and continuous delivery approach

CICS TS open beta

CICS policies

Ansible and the z/OS core and CICS collections

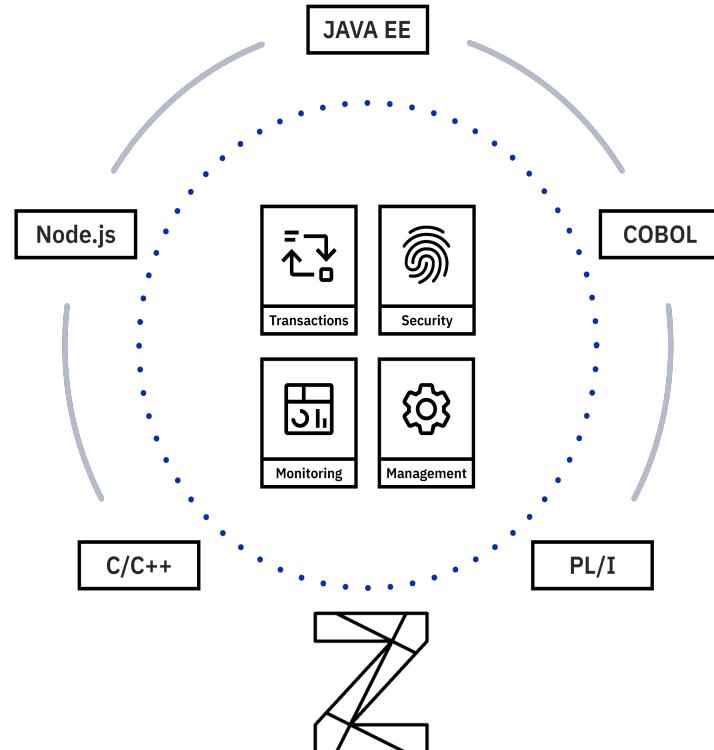


CICS Transaction Server

IBM CICS Transaction Server for z/OS has evolved to over 50 years become a powerful mixed language application server.

Applications can share core programming contexts such as **transactionality, security, monitoring and management**, regardless of the language its components are written in.

Developers can create incredible mixed language applications, that include **Jakarta® Enterprise Edition 8, Spring Boot, Eclipse MicroProfile** and **Node.js®**, together with traditional complied languages like COBOL, C/C++, and PL/I, and Assembler with first-class interoperability. They have access to APIs to access most data and messaging systems, and utilize full power of the IBM Z and z/OS platform.



CICS TS continuous delivery approach

Continuous delivery

- Separate APAR for each feature
- Flexibility of when to apply APAR or to roll back, until APAR become part of pre-req chain. Feature toggles used to selectively opt-in to some new capabilities
- For latest release - CICS TS 5.6 – and previous releases if demand
- CICS Explorer - CD features included with service in CICS Explorer 5.5 update

CICS TS open beta

- Early view of our next version
- Includes CD features and fixes from previous releases
- New features that cannot be made available via service - for example they not complete, or could be disruptive to other products or applications
- Some features may be removed
- We similarly have an open beta version of CICS Explorer

CICS TS continuous delivery approach

New version

- Every 1.5 to 2 years
- 3 editions – MLC, Value Unit Edition, Developer Trial
- *Enhance lifecycle*, meaning 5+ years in service, plus up to 3 years of paid-for extended service

Other

- New capability release elsewhere to improve DevOps and automation scenarios
- For example, Ansible CICS Collection, and plug-ins for Maven, Gradle, UCD

Announcements

- [Announcements](#) of CD features and open betas at regular quarterly intervals

CICS TS 5.6 continuous delivery

New features available via service (APARs) or separate download / plug-in

CICS TS V5.6 [announced](#) in June 2020

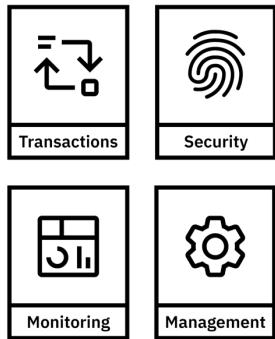
- Features added January 2021 - [announcement](#)
- Features added April 2021 - [announcement](#)

For a consolidated list of all new features, see the CICS TS 5.6 documentation:

- [What's New?](#) - includes a list of features backported to previous releases
- [Changes between releases](#)

Look out for next continuous delivery in November!

CICS TS 5.6 - Foundation



665+ customer requirements addressed
in CICS TS version 5

- New policy system rules, and actions to set CICS region Workload Management health
- Protection against going short-on-storage (SOS) in z/OS 24-bit storage
- Support for IBM z/OS Workload Interaction Correlator to collect top transaction activity at five-second intervals
- Simplified CICS upgrade process and documentation
- Web services enhancements
- Exploit new RACF support of Java Web Token (JWT) for multi-factor authentication

75+ customer requirements addressed
in CICS TS 5.6

CICS TS 5.6 - Foundation

Jan 2021

- Enhanced monitoring for temporary storage capacity
- Resource definition overrides
- New scoping for policy task rules

April 2021

- Improved usage of BAS data space storage for large CICSplex environments
- Support for passing XID to Db2
- Enhanced adapter tracking for CICS applications using Db2
- New policy task rule for container storage

CICS TS 5.6 - Application Development



- New support for Spring Boot
- New support for Jakarta EE 8
- Enhanced CICS Java API
- New Maven Central libraries for CICS Java application development
- New plug-ins for Gradle and Maven to automate building CICS bundles
- New deployment API to simplify CICS bundle deployment during development
- Support for COMMAREAs up to 32 KB on distributed program links

CICS TS 5.6 - Application Development



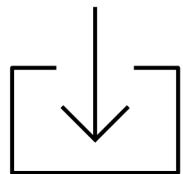
[Jan 2021](#)



- Support for Node.js 14
- Enhanced Zowe CLI CICS deploy plug-in
- New articles and samples to get started with Spring Boot Java applications in CICS
- CICS TS plug-in for UrbanCode Deploy to support CICS TS 5.6

IBM CICS Transaction Server for z/OS V5.6

Upgrading



- New features for developers to modernize CICS applications
- New features for system programmers to better manage and secure CICS
- Opportunity for better performance, and resilience to reduce costs
- Upgrade directly from any release to the latest
- Includes service fixes from previous releases

IBM CICS Transaction Server 5.6

CICS TS 5.6 and continuous delivery approach

CICS TS open beta

CICS policies

Ansible and the z/OS core and CICS collections



CICS TS open beta

CICS TS open beta updated roughly every 3 months for customers and business partners to try out new features and documentation and give feedback

Collaboration with customers

- Request for Enhancements
- CICS Design Partnership

IBM early programs

- Are free of charge
- May not be used for productive purposes
- Not warranted

We'd love to hear your feedback

- Learn more in the [announcement](#) and [What's new?](#)
- Learn even more from CICS developers at the [CICS TS open beta education](#)
- Try it out now by [downloading](#)

CICS TS open beta

Try out the open beta and give us feedback!

- Read the [CICS TS open beta documentation](#)
- Download [CICS TS open beta code](#)
- Raise new [Requests for enhancement](#)

Developer experience and DevOps

- Java annotation now available to define methods to link to OSGi Java applications
- Bundle deployment, Multi-Factor Authentication, and advanced CICS Explorer functions supported in single CICS regions



CICS TS open beta

Security

- Improved security documentation based on scenarios
- New checks for IBM Health Checker for z/OS
- MFA sign-on from CICS Explorer to a single CICS region
- New messages associated with authorization errors to assist in identifying the end user
- New protection to guard against executing code in data-only memory
- New support for transport layer security (TLS) 1.3 and enhanced TLS monitoring
- Removed need for category 1 security definitions



CICS TS open beta



Management

- New system rule type for transaction dump threshold
- Increased capacity of shared data tables
- Extended short on storage notification
- Enhanced performance monitoring
- Enhanced TCP/IP stack diagnostics
- Ability to inquire after 64-bit storage belonging to a task
- Support for daisy-chaining of non-terminal-related START requests



CICS TS open beta



Management

- CICS installation using z/OSMF Software Management
- Use of an alternate Liberty installation location
- Limit on concurrent TLS handshakes
- START CHANNEL support for NOCHECK and PROTECT options



CICS group on the IBM Z and LinuxOne Community

IBM Community

IBM Z and LinuxONE Community Participate Topic groups User groups Solutions Resources

CICS

The world's leading application server software for IBM Z – serving 1.2 million transactions per second

Login or register

Home Blog entries Discussions Events Videos Library Members

This group Related Start a discussion

Blog entries RSS

Using OpenID Connect with CICS Liberty

CICS | Posted by Eric Meier on 03/12/2020

OpenID Connect and JSON Web Tokens (JWTs) are trendy security technologies that many companies choose to use. And believe it or not, you can also use these standards to secure access to CICS applications. In the first part of this blog we'll introduce...

CICS IBM Z OS IBM Z Security Software

On the Move: Data Migration from VSAM to Db2

CICS | Posted by Nick Garner on 03/12/2020

Accelerate and simplify data migration from VSAM to Db2 without the need for costly and complex application rewrites, with CICS VSAM Transparency for z/OS. Why would you want to move data to DB2? Data is valuable, and it's important to maximise this value...

CICS DB2 IBM Z OS IBM Z Software

CICS Explorer, z/OS Explorer, and CICS TS 5.6

CICS | Posted by Dan Holtz on 09/12/2020

CICS Explorer is quite tolerant of versions of CICS TS. To coincide with the release of CICS TS 5.6, we've made it even more so. Read on to find out more! CICS Explorer, z/OS Explorer, and Eclipse: a recap. CICS Explorer provides CICS-specific function...

ADPF CICS IBM Z Hardware IBM Z OS IBM Z

TLS 1.2 session ID caching for CICS in a sysplex

CICS | Posted by Ian Michal on 09/02/2020

What you need to know about TLS session ID caching in a sysplex and why it is important. Every secure, high-value web application which uses TLS 1.2 or earlier would likely collapse under the burden of encryption processing were it not for TLS session IDs.

CICS IBM Z OS IBM Z Security Software

Using MQMONITORS to simplify the administration of CICS-MQ trigger monitors and MQ message consumers

CICS | Posted by Shuan Cao on 08/12/2020

To support the needs of applications that access IBM MQ, CICS TS 5.4 introduces the MQMONITOR resource which provides easier control over CICS-MQ trigger monitor or MQ message consumer tasks. Function overview Capabilities of an MQMONITOR Specify...

CICS IBM Z MQ Security Software

Start a discussion This group Related

How to see what MAPs are in a MAPSET

CICS | Posted by Gregory Odeon on 01/20/2021

CICS IBM Z Software

Coming soon: CICS Technical Exchange

CICS | Posted by Nick Garner on 12/22/2020

CICS Cloud IBM Z

CICS Opportunity in Mexico City

CICS | Posted by Ricardo Garcia on 07/17/2020

CICS IBM Z Software

Welcome to the CICS topic group ...

CICS | Posted by Ben Cox on 07/12/2020

CICS IBM Z Software

All discussions →

<https://www.ibm.com/community/z/>

Samples on GitHub

The screenshot shows the GitHub profile for the 'cicsdev' organization, specifically the 'CICS Transaction Server for z/OS' repository. The page features a dark theme with several pinned repositories displayed prominently. These include:

- cics-java-liberty-springboot-jcics**: A Java Spring Boot application using the JCICS TSQ Java API.
- cics-async-api-redbooks**: Companion code for the CICS Asynchronous API Redbooks publication.
- cics-java-jcicsx-samples**: Samples demonstrating various scenarios for using the JCICSX API.
- cics-nodejs-invoke**: Sample Node.js applications using the invoke API from the ibm-cics-api module.
- cics-java-liberty-springboot-jdbc**: Java Spring Boot application demonstrating the use of the Spring Boot JdbcTemplate in CICS Liberty.

Below the pinned repos, there's a section for 'Repositories' containing one item:

- cics-java-liberty-mq-jms**: Sample Java EE web application demonstrating JMS Connection Factory and MDB usage in CICS Liberty.

On the right side of the page, there are sections for 'People' (showing a grid of user profiles), 'Top languages' (Java, COBOL, Assembly, C, Shell), and 'Most used topics' (java, liberty, blog, javaee, cobol).

github.com/cicsdev

CICS TS online video education courses

Already one of the most popular IBM Redbook deliverables – with over 30,000 unique views.

Course introduction

- ▶ What is CICS?
- ▶ What is an application server?
- ▶ Why use an application server?



Introduction to CICS

This course introduces CICS, an unparalleled mixed language application server. We explore what an application server is, what it provides to developers and what distinguishes CICS as a mixed language application server.

[Open video course](#)

Developing a RESTful Web application for Liberty in CICS

This course provides a step-by-step guide for creating a simple RESTful Java application, then deploying it into WebSphere Liberty running in IBM CICS Transaction Server.

[Open video course](#)

Architecting Java solutions for CICS

This course presents the compelling reasons for developing Java applications in CICS Transaction Server. The course covers the various usage models of Java applications in CICS, along with an explanation of the technologies involved.

[Open video course](#)

Extending a CICS web application using JCICS

This course provides Java application developers with a step-by-step guide to CICS Java development, demonstrating how to access common CICS resources using the JCICS API.

[Open video course](#)

ibm.biz/cics_videocourses

IBM CICS Transaction Server 5.6

CICS TS 5.6 and continuous delivery approach

CICS TS open beta

CICS policies

Ansible and the z/OS core and CICS collections



What is a CICS policy?

A CICS policy has one or more rules each with condition(s) and an action.

A CICS policy is a standard and modern method with real time monitoring of the status of CICS region and CICS tasks.

task.policy

Rules

Rules

type filter text

cont1 (Container storage)
storage (Storage allocation)

General Information

Rule type: Storage allocation
Perform an action when the amount of storage that is allocated by a user task exceeds a threshold. The threshold applies to a specific storage class and is not a cumulative count of all storage requests.

Description:

Condition

This rule will trigger when the following condition is met:

All storage allocated 1

Limit this rule to specific transaction IDs and user IDs:

Transaction ID:

User ID:

This rule requires CICS TS 6.1 or later

Action

What action should be taken when the threshold is exceeded?

Issue a message
 Emit an event:

EP Adapter
Event name:

Static Data (0 items)

Abend the task with abend code AMPB

Overview Rules

When to use CICS policies?

Care about CICS region's resource status change?

- CICS policy system rules monitor system resource status
- Action for system rules can be a message, a CICS event, or changing z/OS WLM open status
 - For AID threshold rule, can be rejecting an EXEC CICS request

Care about excessive storage or amount of API commands by a user task?

- CICS policy task rules monitor resource usage by individual task
- Action for task rules can be a message, a CICS event or abend the task

CICS policy rules monitor the system in real time, with no-polling mechanism

What can I monitor on system resources?

CICS TS 5.4

- Bundle available status
- Bundle enable status
- DB2 connection status
- File enable status
- File open status
- IPIC connection status
- MRO connection status
- Program enable status
- Message
- Unhandled transaction abend
- Transaction class tasks
- User Tasks

CICS TS 5.5

- DBCTL connection status
- IBM MQ Connection status
- Pipeline enable status
- AID threshold

CICS TS 5.6

- Transaction dump threshold (APAR PH34348, CICS Explorer 5.5.0.17)

CICS TS 6.1 open beta

- Compound condition

What can I monitor on individual tasks?

CICS TS 5.4

- ASYNC requests
- Database requests
- EXEC CICS requests
- File requests
- IBM MQ requests
- Named counter requests
- Program requests
- Start requests
- Syncpoint requests

- Storage allocation
- Time (CPU, elapsed)
- Storage requests
- TS queue bytes
- TS queue requests
- TD queue requests
- Transaction id and user id
(APAR PH26145, CICS Explorer 5.5.0.11)

CICS TS 5.6

- Container storage
(APAR PH29187, CICS Explorer 5.5.0.15)

CICS TS 6.1 open beta

- Ability to specify ALL for these conditions
- Storage requests
- TS queue requests
- TD queue requests
- File requests

CICS policies vs. CICS application events

CICS policies

- Use rules to monitor system resource status, the amount of resource usage and unusual system states
- Action can be a message, a CICS event, abending task, changing z/OS WLM status, or reject EXEC CICS request

CICS system events are deprecated
and replaced by CICS policies

CICS application events

- Use capture specifications to filter on application data on selected EXEC CICS API calls
- Action is a CICS event

Compound condition rule – CICS 6.1 open beta

CICS region opens to business when ALL of the defined conditions are satisfied, for example:

- When certain bundles are enabled
- AND when certain connections are connected
- AND when certain files are opened and enabled
- ...

Open the region's z/OS WLM status (CICS 5.6), noticed by:

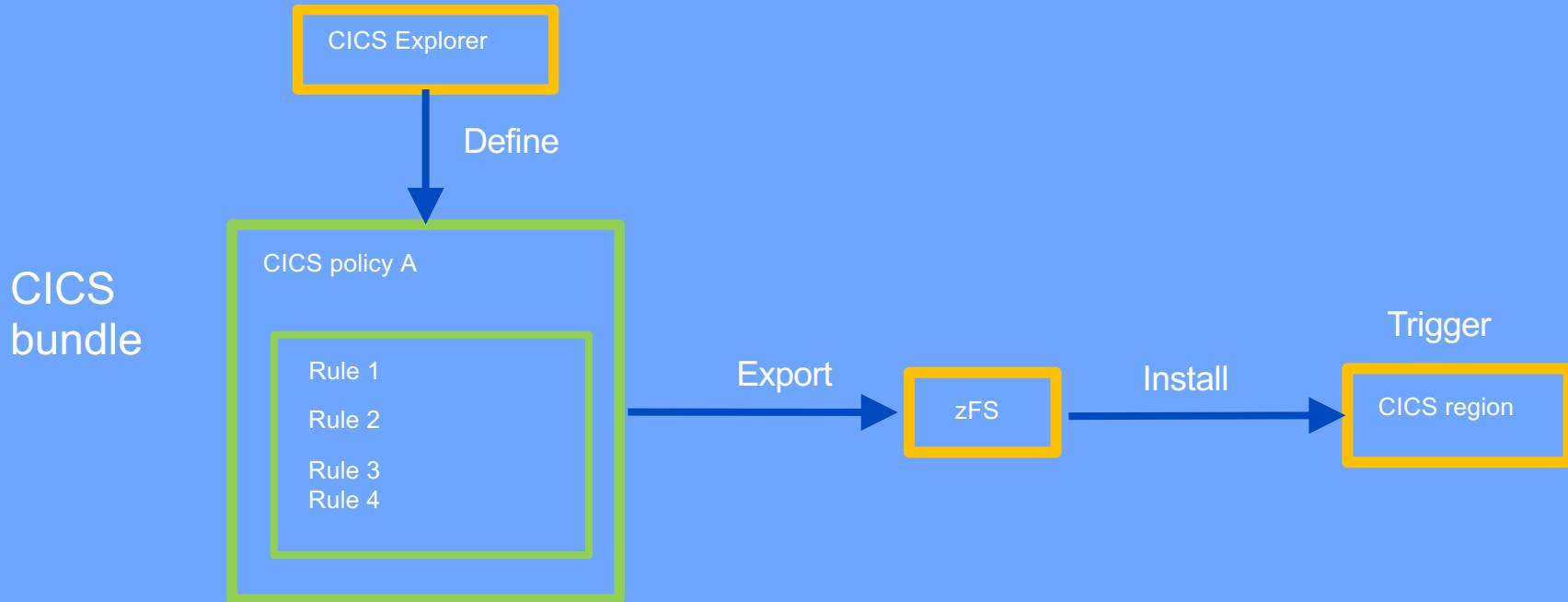
- TCP/IP
- CICSplex SM
- MQMONITORs (CICS-MQ component)

CICS checks conditions when a compound condition rule is enabled and monitors them afterwards

Condition types

- Bundle available status
- Bundle enable status
- DB2 connection status
- File enable status
- File open status
- IPIC connection status
- MRO connection status
- Program enable status
- DBCTL connection status
- IBM MQ Connection status
- Pipeline enable status

- Watch demo at [CICS Technical Exchange 30th September 2021](#)
- Defining a CICS policy and a group of rules
- Installing the CICS policy
- Triggering the rules to manage CICS system WLM health status



IBM CICS Transaction Server 5.6

CICS TS 5.6 and continuous delivery approach

CICS TS open beta

CICS policies

Ansible and the z/OS core and CICS collections



Agenda

Today we'll be looking at **Ansible** for z/OS and especially how that relates to CICS Transaction Server.

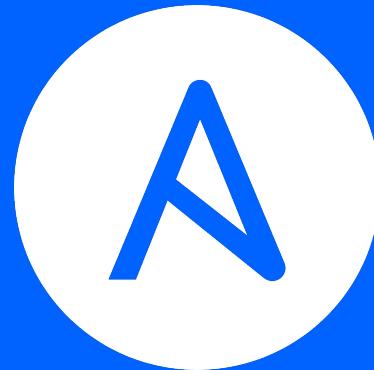
- What is Ansible and what support does it have on z/OS?
- Introduction to the CICS collection for Ansible

About Ansible

Ansible is a provisioning, configuration management and application deployment tool.

Tagline: “Turn tough tasks into repeatable playbooks”.

Rather than managing one system at a time, Ansible models your IT infrastructure by describing how all your systems inter-relate.a



Common usage of Ansible

- System provisioning
- Installing applications
- Managing users
- Updating certificates
- Continuous delivery
- Configuration management



ANSIBLE

Why is Ansible so popular?

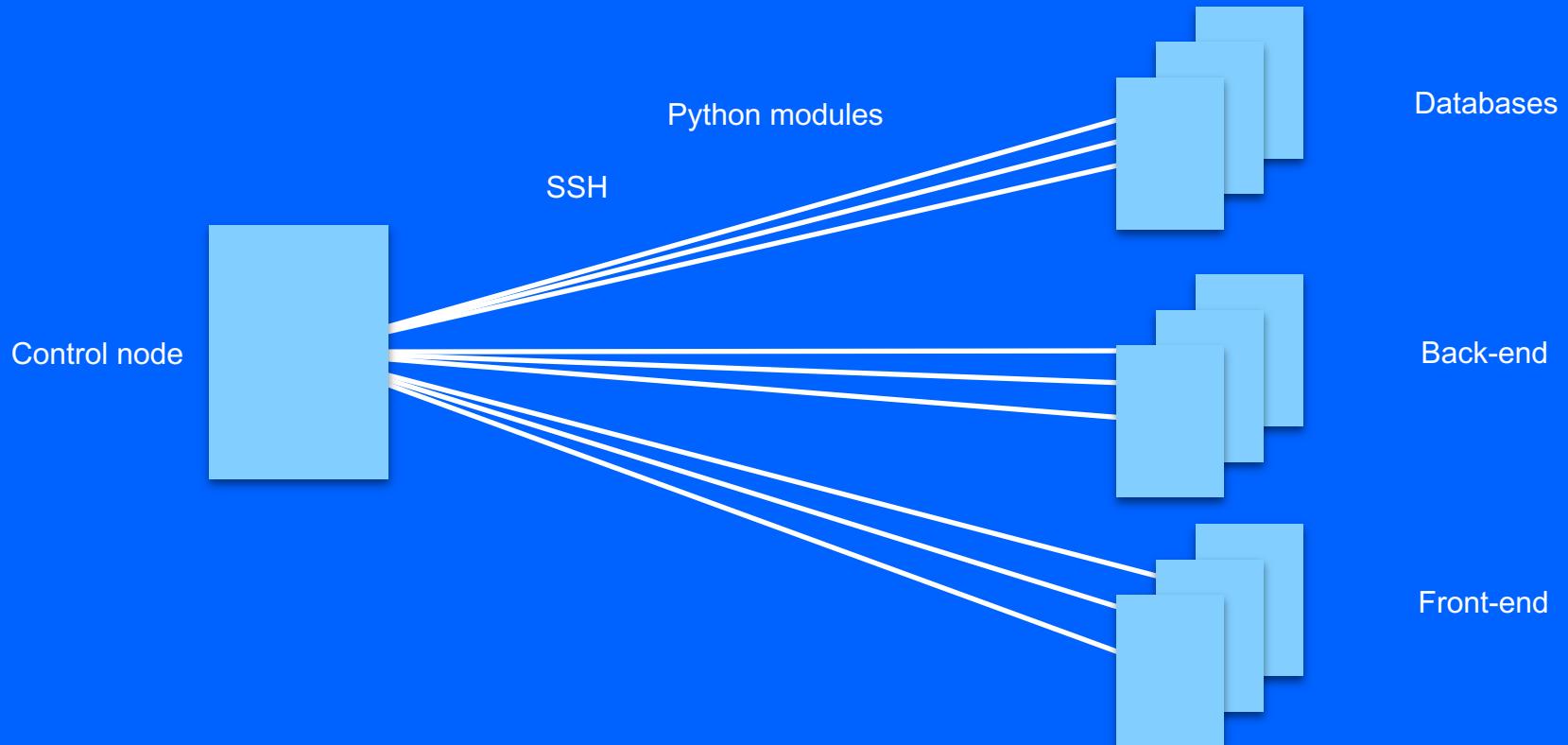
Ansible is extremely popular across many enterprises. For example, it's now the top cloud configuration tool and is heavily used on-prem too.

Some of the reasons for that include:

- It normalizes tooling across a multitude of platforms
- It centralizes your enterprise automation strategy
- You can achieve configuration as code
- Over 3000 modules for all the things you might need it to do

If you're not aware of it already in your enterprise, ask around!

The basic concepts of Ansible



The Ansible inventory

The inventory identifies the nodes that are managed, and categorises them into different groups.

Both nodes and groups can be assigned variables for use later during automation.

Inventory can be static (via a file) or dynamically provided.

```
back-end.example.com
```

```
[databases]
```

```
db1.example.com
```

```
db2.example.com
```

```
db3.example.com
```

```
[front-end]
```

```
fe1.example.com
```

```
fe2.example.com
```

```
fe3.example.com
```

Ansible playbooks

Playbooks bind hosts from the inventory to tasks

The hosts can be individual nodes, groups or everything

When a playbook is executed, it runs tasks against the hosts they're bound to

```
---
- name: update web servers
  hosts: webservers
  remote_user: root

  tasks:
    - name: ensure apache is at the latest version
      yum:
        name: httpd
        state: latest
    - name: write the apache config file
      template:
        src: /srv/httpd.j2
        dest: /etc/httpd.conf

- name: update db servers
  hosts: databases
  remote_user: root

  tasks:
    - name: ensure postgresql is at the latest version
      yum:
        name: postgresql
        state: latest
    - name: ensure that postgresql is started
      service:
        name: postgresql
        state: started
```

Ansible tasks

Tasks are work that you configure to be run on the managed nodes.

Some are built into Ansible, but many are provided by Ansible plugins.

Tasks are configured using Ansible, with configuration that is relevant to that task.

```
- shell:  
  chdir: somedir/  
  cmd: ls -l | grep log
```

Running a playbook

To run Ansible, it needs to be installed on a control node.

The playbook can then be run from the command line.

The tasks etc — composed of Python modules — are shipped from the control node to the managed nodes.

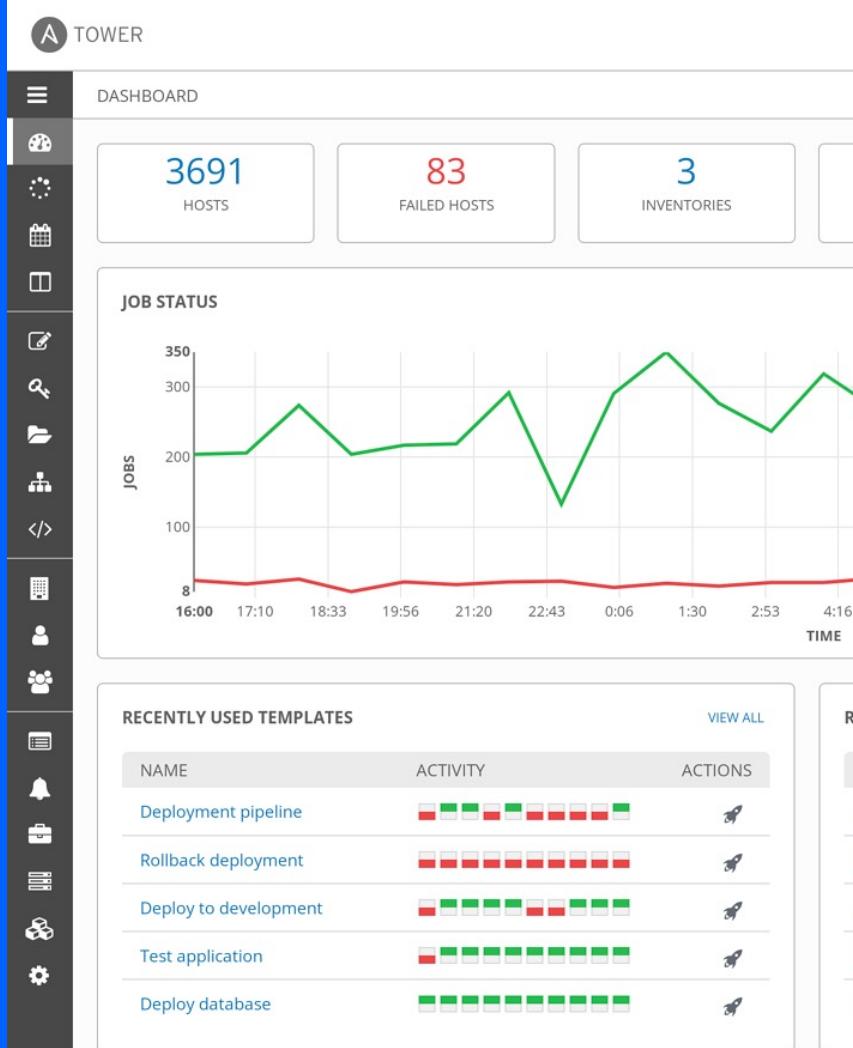
```
ansible-playbook  
  -i databases,  
  -e varname=varval  
  my_playbook.yml
```

Automation as a service

By using Ansible Tower across the enterprise you can take a higher-level view, without needing to know the specifics of exactly how individual playbooks are put together.

Users can use Tower to run playbooks without having to install Ansible themselves, and run them with functional credentials.

They can also schedule automation to run at specific times, such as a monthly audit or a dashboard refresh every minute.



Why would you want to use Ansible on z/OS?

Using Ansible on z/OS allows you to centralise your automation skill set around a particular open source technology that gives you flexibility and power.

By sharing the same automation strategy as the rest of the enterprise you can unlock opportunities for collaboration and integration.

Ansible's flexibility permits reuse of your existing automation — triggering System Automation (SA), z/OSMF workflows, JCL, ... — or adaptation to specific Ansible tasks.

Ansible for z/OS

There is a growing set of collections that support z/OS function on [Ansible Galaxy](#) — the repository for Ansible collections of tasks.

In addition, they're also part of the Red Hat Ansible Certified Content for IBM Z. This means they're available on [Ansible Automation Hub](#) and are fully supported by Red Hat and IBM.

The `ibm_zos_core` collection gives a good foundational experience on z/OS around datasets, APF, transferring data, operator commands, and so on.

Collections 4

`ibm_zos_core`

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

16 Modules 0 Roles 20 Plugins



ibm z zos z_os core zos_core ibm_zos_core data_set jcl uss mvs

`ibm_zos_ims`

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

7 Modules 0 Roles 8 Plugins



ibm z zos z_os core zos_core data_set jcl uss mvs ims zos_ims

`ibm_zos_sysauto`

The IBM Z System Automation collection includes roles and sample playbooks to access the IBM Z System Operations REST server.

0 Modules 2 Roles 0 Plugins



ibm z zos z_os zos_sa ibm_zos_sa system_automation

`ibm_zos_zosmf`

Ansible collection consisting of modules and roles to work with z/OS based on z/OS Management Facility (zosmf).

7 Modules 6 Roles 6 Plugins



ibm z zos zosmf zos_management_facility mvs workflow job console cpm cloud_provisioning cics db2 ims mq liberty was dataset uss

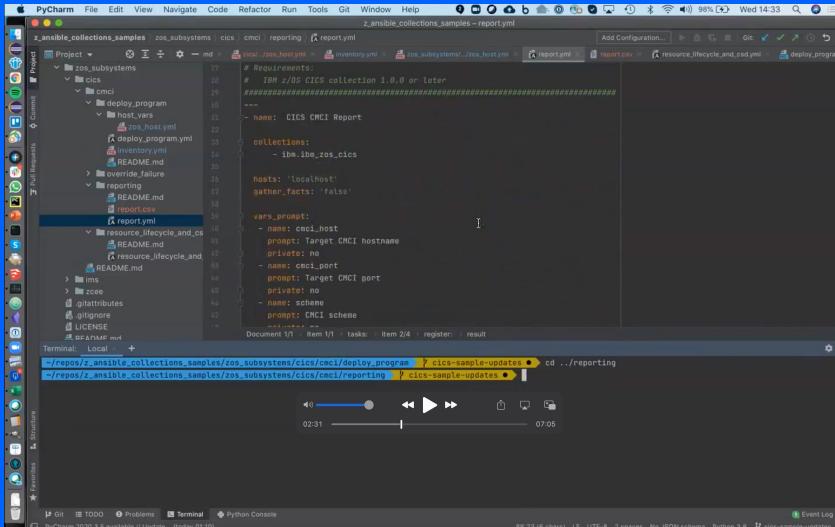
The CICS TS collection for Ansible

We're in the process of developing a CICS collection for Ansible — bringing tasks that are commonly required when working with CICS TS.

It's being developed in the open at github.com/ansible-collections/ibm_zos_cics.

The collection is in Ansible Galaxy at galaxy.ansible.com/ibm/ibm_zos_cics.

The collection will be part of the (supported) Red Hat Ansible Certified Content for IBM Z.



```
# Requirements:
# IBM i/OS CICS collection 1.0.0 or later
#
# --- name: CICS CMCI Report
#
# collections:
#   - ibm.ibm_zos_cics
#
# hosts: 'localhost'
# gather_facts: 'false'
#
# vars_prompt:
#   - name: cmci_host
#     prompt: Target CMCI hostname
#   - name: cmci_port
#     prompt: Target CMCI port
#   - name: scheme
#     prompt: CMCI scheme
#
# Document V1 | Rem V1 | Tasks | Item 2/4 | register: result
```

Terminal Local ↗
~/repo/z ansible_collections_samples/zos_subsystems/cics/cmci/deploy_program ↗ cics-sample-updates ↗ cd ../reporting
~/repo/z ansible_collections_samples/zos_subsystems/cics/cmci/reporting ↗ cics-sample-updates ↗

Demo

Also at [CICS Technical Exchange](#) [21st April 2021](#) at 29:54 mins into the recording to 36:55

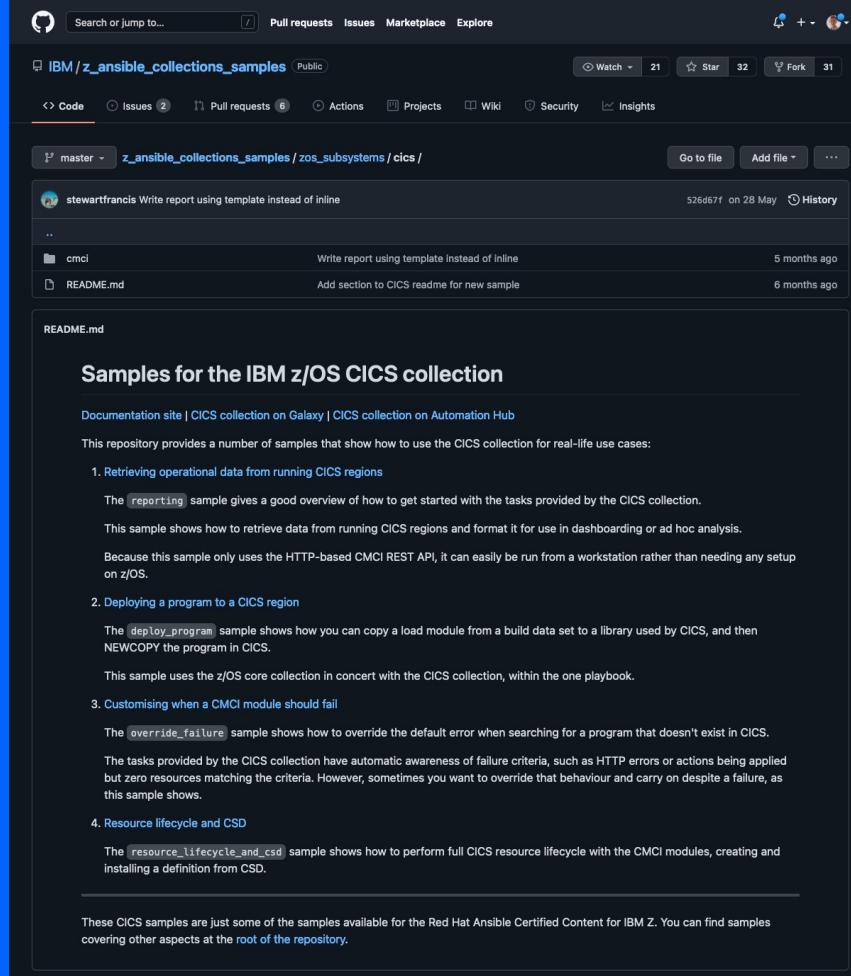
Tasks in the CICS collection

The CICS collection includes the following tasks:

- cmci_action
- cmci_create
- cmci_delete
- cmci_get
- cmci_update

All use CMCI (the HTTP API to manage CICS) so can be run on the target node (the z/OS LPAR) or remotely.

CMCI is the same API that Explorer uses. You can use our Ansible collection to automate lots of things you might do manually today in Explorer.



The screenshot shows a GitHub repository page for `IBM/z_ansible_collections_samples`. The repository has 21 stars and 32 forks. The `zos_subsystems/cics` directory contains two files: `cmci` and `README.md`. The `cmci` file is described as "Write report using template instead of inline" and was last updated 5 months ago. The `README.md` file is described as "Add section to CICS readme for new sample" and was last updated 6 months ago. Below the files, there is a section titled "Samples for the IBM z/OS CICS collection" which lists four samples: 1. Retrieving operational data from running CICS regions, 2. Deploying a program to a CICS region, 3. Customising when a CMCI module should fail, and 4. Resource lifecycle and CSD. Each sample is described with a brief overview and a link to the corresponding file in the repository.

Search or jump to... Pull requests Issues Marketplace Explore

Code Issues 2 Pull requests 6 Actions Projects Wiki Security Insights

master z_ansible_collections_samples / zos_subsystems / cics

stewartfrancis Write report using template instead of inline 526d67f on 28 May History

.. cmci Write report using template instead of inline 5 months ago

README.md Add section to CICS readme for new sample 6 months ago

README.md

Samples for the IBM z/OS CICS collection

Documentation site | CICS collection on Galaxy | CICS collection on Automation Hub

This repository provides a number of samples that show how to use the CICS collection for real-life use cases:

- 1. Retrieving operational data from running CICS regions**
The `reporting` sample gives a good overview of how to get started with the tasks provided by the CICS collection. This sample shows how to retrieve data from running CICS regions and format it for use in dashboarding or ad hoc analysis. Because this sample only uses the HTTP-based CMCI REST API, it can easily be run from a workstation rather than needing any setup on z/OS.
- 2. Deploying a program to a CICS region**
The `deploy_program` sample shows how you can copy a load module from a build data set to a library used by CICS, and then NEWCOPY the program in CICS. This sample uses the z/OS core collection in concert with the CICS collection, within the one playbook.
- 3. Customising when a CMCI module should fail**
The `override_failure` sample shows how to override the default error when searching for a program that doesn't exist in CICS. The tasks provided by the CICS collection have automatic awareness of failure criteria, such as HTTP errors or actions being applied but zero resources matching the criteria. However, sometimes you want to override that behaviour and carry on despite a failure, as this sample shows.
- 4. Resource lifecycle and CSD**
The `resource.lifecycle_and_csd` sample shows how to perform full CICS resource lifecycle with the CMCI modules, creating and installing a definition from CSD.

These CICS samples are just some of the samples available for the Red Hat Ansible Certified Content for IBM Z. You can find samples covering other aspects at the [root of the repository](#).

Getting started

To get going, go to the [CICS collection on Ansible Galaxy](#), and follow the instructions to install Ansible and the collection.

Separately, ensure you have enabled CMCI on your target CICS regions or CICSplexes.

Use our ['reporting' sample](#) to run your first playbook with the CICS collection.

Thank you!

Any questions?

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.

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.

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.
- 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.

Notices and disclaimers

- 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, ibm.com and [names of other referenced IBM products and services used in the presentation] 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

