
IBM zSystems

FAQs

July 2022

IBM z/OS Change Tracker

Frequently Asked Questions

Worldwide

IBM

Table of Contents

General and Value 3

Technical 5

Dependencies and Security 9

General and Value

What is IBM z/OS® Change Tracker, and why should I use it?

IBM z/OS Change Tracker is a System Software Change Control and Management solution for the z/OS platform. System programmers can identify, track, and recover hundreds of configuration files in real-time using IBM z/OS Change Tracker. Mission-critical system and application software configuration data can be monitored automatically in real-time for z/OS environments. At a glance, IBM z/OS Change Tracker offers:

- **Unprecedented control:** Track and control real-time system-wide software configuration changes.
- **Enhanced recovery processes:** Automatic versioning of identified system control data sets.
- **Rollback Capabilities:** Rollback capabilities to undo unplanned/unsuccessful promotions.
- **Next-level monitoring:** Data set recovery and reporting of real-time configuration data changes.

Why is IBM z/OS Change Tracker only available on z/OS V2.5? What about z/OS V2.4?

IBM z/OS Change Tracker is only available on z/OS V2.5 due to functional dependencies which are only satisfied with that release. Those functional dependencies were not provided in z/OS releases prior to z/OS V2.5, meaning that IBM z/OS Change Tracker is available for V2.5 and above.

What added value can the priced feature IBM z/OS Change Tracker bring to my system?

Although you may have existing functions (or indeed, other vendor products which you pay for today) which supply similar capabilities, we feel that IBM z/OS Change Tracker is superior at providing additional capabilities that z/OS Systems Programmers need for robust and real-time systems management.

For instance, you might have the existing ability to determine who has modified a certain data set or have basic security protection for an entire data set, but IBM z/OS Change Tracker can extend further into that capability and allow an administrator to allow specific check in and check out permission such that modifications can be only performed at certain times. Further, an enterprise can determine that any modifications require an explanation provided for the modification which is saved with the change, and then be notified of changes that occurred. These easily performed functions, along with others, make IBM z/OS Change Tracker more valuable to the z/OS Systems Programmer.

Do system programmers require extensive knowledge of z/OS to be able to utilize IBM z/OS Change Tracker?

No, IBM z/OS Change Tracker helps to replace manual procedures for auditing and investigating systems changes. using concepts that are easy to understand and follow. For

instance, you could review the information collected by IBM z/OS Change Tracker to determine what system configuration changes have occurred during the course of a system failure investigation. Then, be able to do recovery from an unplanned or undesired change quickly. Member-level backups generated by IBM z/OS Change Tracker when the change has been made are available. This allows z/OS Systems Programmers and system auditors to easily pinpoint what, who, and how changes were made.

Where can I find out more information about IBM z/OS Change Tracker, and see what it looks like? Where can I get started?

Visit <https://www.ibm.com/support/z-content-solutions/zos-change-tracker/> for more information. Also found on this web page are links to a presentation with detailed screen shots of IBM z/OS Change Tracker use cases, and the official product publication link.

Technical

How do I install IBM z/OS Change Tracker?

IBM z/OS Change Tracker was made generally available on May 13, 2022. However, the inclusion into z/OS V2.5 ServerPacs was available later due to the customization required for IBM ServerPac®. If you want to use IBM z/OS Change Tracker and you have not received it with your z/OS V2.5 ServerPac, the easiest method is to order a z/OS V2.5 CBPDO, indicating on Shopz that you wish to be entitled to IBM z/OS Change Tracker. (*You can easily check your z/OS V2.5 ServerPac to see if it has been installed by seeing if SMP/E FMID HCYG100 is installed.*)

If you order a z/OS V2.5 CBPDO with IBM z/OS Change Tracker, you will be delivered the entire z/OS V2.5 product, along with IBM z/OS Change Tracker. Don't worry, you don't need to install all of z/OS V2.5! The IBM z/OS Change Tracker is a very simple install of SMP/E FMIDs (*HCYG100 base, and JCYG10J for Japanese*), using the supplied z/OS V2.5 Program Directory. IBM z/OS Change Tracker installs into its own data sets and sample jobs are supplied.

How do I configure IBM z/OS Change Tracker?

IBM z/OS Change Tracker, like many other functions, is configured with a z/OSMF Workflow. You can find the supplied Workflow in *path-prefix/usr/lpp/cyg/zosmf/workflow/cygwflw.xml* after you have completed your installation.

Now that I have IBM z/OS Change Tracker configured and up and running, how do I start tracking my data sets?

Once you have the IBM z/OS Change Tracker started task up and running and your IBM z/OS Change Tracker administrators and users defined, you can start identifying the resources you wish to track and the desired characteristics of them. A good place to start is the 'Administration' section of the *Guide and Reference* found here: <https://www.ibm.com/docs/en/zos/2.5.0?topic=ispf-administration>.

I see in the *Guide and Reference* that IBM z/OS Change Tracker has an ISPF panel interface with batch jobs that can be run. Don't you have a z/OSMF interface?

IBM z/OS Change Tracker does have an ISPF interface and batch jobs that can be run, however we are also planning a z/OSMF Change Tracker plugin. The z/OSMF plugin is designed to offer a modern graphical user interface that will enable a more robust and simpler user experience with IBM z/OS Change Tracker. By using recognizable icons and intuitive compare methods, customization and reporting from IBM z/OS Change Tracker is planned to identify resources to manage, check in and check out data sets and members, and is designed to be quickly understandable and lessen time to productive use.

I see there are lots of functions in IBM z/OS Change Tracker which help the z/OS Systems Programmer, but there are other helpful functions out there. How does IBM z/OS Change Tracker fit with those other solutions?

There certainly are a lot of z/OS functions and IBM products available to help z/OS Systems Programmers perform their job! And you probably have some homegrown tools as well. IBM z/OS Change Tracker is not intended to replace any of those, but rather complement and augment them by providing additional capabilities that are needed to help track configuration data sets to keep your systems running with the configuration that you intend.

For instance, specific component knowledge has been encoded into IBM Health Checks to help advise on preferred configurations or an upcoming upgrade action that you will encounter. IBM z/OS Change Tracker does not interfere with component verification during its tracking of your configuration files, managing backups, and controlling who can and cannot access resources at a certain time. If you change your configuration based on the advice of an IBM Health Check, IBM z/OS Change Tracker can take a backup of that change automatically, identify who/how/when the change was done, and even require a reason for doing that permitted change (so the user can indicate it was a specific IBM Health Check perhaps). This scenario shows that IBM z/OS Change Tracker does not conflict with, but rather can nicely augment a change that was indicated by an IBM Health Check.

Are load libraries supported? What kinds of data set can I track?

You can monitor a load library if you desire. Today, IBM z/OS Change Tracker can monitor and protect PDS, and PDSE data sets. In addition, sequential, and entire VSAM data sets (including entire zFS data sets) and volumes can be identified for comparisons in snapshots. We have a requirement for being able to monitor individual z/OS UNIX® files. If you need other types of data set monitored, please open a requirement (now called an "Idea") as we are interested in making sure that IBM z/OS Change Tracker provides comprehensive coverage of your system configuration assets.

How does the IBM z/OS Change Tracker differ from the z/OS Generic Tracking Facility?

The z/OS Generic Tracking Facility is intended to surface desired information when programs are run and an event is detected which is important to note. The Generic Tracker captured events you might be familiar with, such as the use of one-byte console IDs, specific JES statements, or DFSMS™ EAV information. The Generic Tracker provides a callable service which applications or products can exploit to surface this information.

IBM z/OS Change Tracker is intended to help the z/OS systems programmer manage system configuration data sets easier.

What is the difference between Fingerprint/Tokenization/Snapshot?

We understand the confusion over these terms and are trying to consolidate them to make it easier to follow:

- The term "snapshot" is the preferred word for the concept of capturing an instance of data at a point in time. IBM z/OS Change Tracker intends to use this term consistently as future functions are delivered.
- The terms "tokenization" and "token" are today used heavily within the ISPF panels and batch JCL for IBM z/OS Change Tracker. This relates to the internal implementation that IBM z/OS Change Tracker uses to capture the instance of data that a user has requested. Although the internal implementation of tokenization is planned to continue in IBM z/OS Change Tracker with future planned functions, you probably will see that term used less frequently in interfaces.
- The term "fingerprint" is planned to be used by IBM z/OS Change Tracker in the z/OSMF interface, rather than the term "tokenization". "Tokenization" and "fingerprint" are the same, but within an intended z/OSMF interface, "fingerprint" is preferred.

Is the “snapshot” comparable to other solutions using checksums?

IBM z/OS Change Tracker uses its own unique method for taking snapshots and has its own internal implementation to capture an instance of data. The method it uses is not disclosed.

How does IBM z/OS Change Tracker differ from the Git Source Code repository?

Let's start with some definitions and use case scenarios to explain the difference.

Git is a version control or library system that lets you manage and keep track of your source code history. As with any library system, you can make source code changes and track who and when these changes happened including the state of those changes (committed or not). The audience for git is application developers, who require a code repository to track source code changes, useful in their application development pipeline. In fact, git is probably the premier source code repository for applications in the DevOps process, evidenced by the multitude of open-source projects that store and manage their source code using git.

IBM z/OS Change Tracker is a system software change control solution that can track, control, and manage configuration changes for the z/OS platform. Think SYS1.PARMLIB or SYS1.PROCLIB, where having a mechanism that allows you to track and control changes to these resources is useful for audit and integrity purposes. The audience for IBM z/OS Change Tracker would be the system programming staff or the audit team that are required to maintain strict controls over changes that would affect system resources.

Git and IBM z/OS Change Tracker share similar concepts (basic functions). For example:

- Comparing files – Git users would use this capability to compare changes made to a source header file; IBM z/OS Change Tracker users would use this capability to compare changes to the BPXPRMxx member of PARMLIB
- tracking changes – Git users would use this capability to see that a change was made by Jane, an application developer, on June 22, 2021 to a source code file; IBM z/OS Change Tracker users would use this capability to see that Zach, a systems programmer, made changes to the IEASYSxx PARMLIB member last month.

But there also differences between git and IBM z/OS Change Tracker. Besides the difference in audience and use-case, there are differences in capabilities. For example:

- Real-time control – IBM z/OS Change Tracker can capture and control live attempts to update actual data sets in order to protect unintended changes, live on the z/OS LPAR; Git can only track changes within its own repository.
- Running reports –IBM z/OS Change Tracker provides the extensive capability to create customized reports useful for system-wide changes; Git has a very rudimentary reporting capability, for instance, a log report showing an author's activity.

Now technically speaking, one could place SYS1.PARMLIB in git to handle updates and approvals using git's semantics. While changes to individual members could be handled via pull requests, there would still need to be a mechanism to reflect these updates back in the data set on the z/OS LPAR. This “mapping” or the copying back and forth from the repo to the data set (however automated), creates unnecessary overhead and still leaves open the need to monitor update attempts to the data set in real-time. Use of Git to track SYS1.PARMLIB changes would not be a good use-case.

All in all, git should really be thought about as application development tooling, while IBM z/OS Change Tracker tends to deliver more what z/OS system programmers have come to expect from the world-class operating system across an entire enterprise.

Dependencies and Security

Are there prerequisites for running IBM z/OS Change Tracker?

The only prerequisite is z/OS V2.5, and two V2.5 PTFs which are indicated during the installation. Since IBM z/OS Change Tracker is a priced feature, then entitlement of the feature is also required which allows the feature to be enabled in your IFAPRDxx parmlib member.

Is DFSMSHsm™ a requirement? If not, what utilities are used for backup?

IBM z/OS Change Tracker has no functional dependency on DFSMSHsm or any other priced z/OS feature or program product. In fact, the only functional dependency is the base z/OS V2.5 itself (and some z/OS V2.5 PTFs already mentioned), and since IBM z/OS Change Tracker happens to be in z/OS V2.5 that means there are no external dependencies. IBM z/OS Change Tracker uses its own internal implementation to keep saved versions of changed resources in its own repository.

What additional security benefit is there compared to what I have with RACF®, Top Secret, or ACF2?

Typically, a security product will control access to a complete data set and the members contained in that data set. IBM z/OS Change Tracker has configuration options that allow a more granular approach to monitoring and controlling access to these members. If a data set is “locked” by the IBM z/OS Change Tracker Administrator, members must first be checked out to a user or group before they can make any updates *even if they have authority to write to the data set*. Any attempts to update a member before being checked out will result in the update failing and can be tracked in the IBM z/OS Change Tracker activity reports.

In addition, the administrator can enforce use of comments to document the reason for the change to that member, as well as taking a backup of the member once it has been changed. This allows a review of both the previous and updated versions of the member, as well as the ability to easily restore to a previous version from the IBM z/OS Change Tracker ISPF panels.

How does IBM z/OS Change Tracker work with my external security manager (like RACF)? Does it replace it?

Your external security manager still controls the access to the data set underneath IBM z/OS Change Tracker. In other words, you can't "bypass" any RACF permissions, as IBM z/OS Change Tracker just has additional granular control on top of your security product.

What are the corresponding RACF profiles / classes?

There are new resource profiles needed for IBM z/OS Change Tracker, and these profiles can be viewed and validated using the z/OSMF Security Configuration Assistant. Use the z/OSMF configuration workflow for IBM z/OS Change Tracker to set them up. IBM z/OS Change Tracker grants different levels of access and involvement depending on user role: administrator, user, or auditor.

- Administrators are able to define libraries to be monitored and protected, and can grant individuals access to update locked members.
- Users can verify that software on multiple local or remote systems is identical. Users can view and compare backups. If IBM z/OS Change Tracker finds any differences, it identifies which members in the libraries have different contents.
- Auditors are participants who are able to receive and view outputs generated by IBM z/OS Change Tracker.



©Copyright IBM Corporation 2022
IBM Corporation
New Orchard Road
Armonk, NY 10504
U.S.A.
07/22

IBM, ibm.com, the IBM logo, DFSMS, DFSMSShm, RACF, ServerPac and z/OS are trademarks or registered trademarks of the International Business Machines Corporation.

A current list of IBM trademarks is available on the Web at <https://www.ibm.com/legal/us/en/copytrade.shtml>, and select third party trademarks that might be referenced in this document is available at https://www.ibm.com/legal/us/en/copytrade.shtml#section_4.

Adobe, the Adobe logo, PostScript, and the PostScript logo are either registered trademarks or trademarks of Adobe Systems Incorporated in the United States, and/or other countries.

Cell Broadband Engine is a trademark of Sony Computer Entertainment, Inc. in the United States, other countries, or both and is used under license therefrom.

InfiniBand and InfiniBand Trade Association are registered trademarks of the InfiniBand Trade Association.

Intel, Intel logo, Intel Inside, Intel Inside logo, Intel Centrino, Intel Centrino logo, Celeron, Intel Xeon, Intel SpeedStep, Itanium, and Pentium are trademarks or registered trademarks of Intel Corporation or its subsidiaries in the United States and other countries.

Java and all Java-based trademarks and logos are trademarks or registered trademarks of Oracle and/or its affiliates.

The registered trademark Linux® is used pursuant to a sublicense from the Linux Foundation, the exclusive licensee of Linus Torvalds, owner of the mark on a worldwide basis.

Microsoft, Windows, Windows NT, and the Windows logo are trademarks of Microsoft Corporation in the United States, other countries, or both.

OpenStack is a trademark of OpenStack LLC. The OpenStack trademark policy is available on the [OpenStack website](#).

Red Hat®, JBoss®, OpenShift®, Fedora®, Hibernate®, Ansible®, CloudForms®, RHCA®, RHCE®, RHCSA®, Ceph®, and Gluster® are trademarks or registered trademarks of Red Hat, Inc. or its subsidiaries in the United States and other countries.

RStudio®, the RStudio logo and Shiny® are registered trademarks of RStudio, Inc.

TEALEAF is a registered trademark of Tealeaf, an IBM Company.

UNIX is a registered trademark of The Open Group in the United States and other countries.

Worklight is a trademark or registered trademark of Worklight, an IBM Company.

Zowe™, the Zowe™ logo and the Open Mainframe Project™ are trademarks of The Linux Foundation.

All statements regarding IBM's future direction and intent are subject to change or withdrawal without notice, and represent goals and objectives only. The information contained in this documentation is provided for informational purposes only. While efforts were made to verify the completeness and accuracy of the information contained in this documentation, it is provided "as is" without warranty of any kind, express or implied. In addition, this information is based on IBM's current product plans and strategy, which are subject to change by IBM without notice. IBM shall not be responsible for any damages arising out of the use of, or otherwise related to, this documentation or any other documentation. Nothing contained in this documentation is intended to, nor shall have the effect of, creating any warranties or representations from IBM (or its suppliers or licensors), or altering the terms and conditions of the applicable license agreement governing the use of IBM software.

References in these materials to IBM products, programs, or services do not imply that they will be available in all countries in which IBM operates. Product release dates and/or capabilities referenced in these materials may change at any time at IBM's sole discretion based on market opportunities or other factors and are not intended to be a commitment to future product or feature availability in any way.