

z/OS
3.2

Infusing AI into z/OS



Note

Before using this information and the product it supports, read the information in [“Notices” on page 95.](#)

This edition applies to IBM® z/OS® 3.2 (5655-ZOS) and to all subsequent releases and modifications until otherwise indicated in new editions.

Last updated: 2025-09-30

© **Copyright International Business Machines Corporation 2023, 2025.**

US Government Users Restricted Rights – Use, duplication or disclosure restricted by GSA ADP Schedule Contract with IBM Corp.

Contents

Figures.....	v
Tables.....	vii
About Infusing AI into IBM z/OS.....	ix
How to provide feedback to IBM.....	xi
Summary of changes.....	xiii
Summary of changes for z/OS 3.2.....	xiii
Summary of changes for z/OS 3.1.....	xiii
Part 1. Overview of the AI Framework for IBM z/OS.....	1
Chapter 1. Introduction to the AI Framework for z/OS.....	3
Useful terminology.....	3
z/OS AI Framework components.....	4
Chapter 2. z/OS AI Framework software and product packaging.....	7
Part 2. Getting started with the AI Framework for IBM z/OS.....	11
Chapter 3. Planning and installation.....	13
Hardware and software requirements.....	13
Default network port usage.....	15
Installing the software.....	15
Chapter 4. Configuring the z/OS AI Framework.....	17
What is a workflow?.....	17
Workflows overview.....	17
Using the AI Framework for IBM z/OS Configuration Workflow.....	17
Adding another z/OS system to your z/OS AI Framework configuration.....	20
Importing and exporting configuration variables in a property file.....	22
Avoiding AT-TLS configuration conflicts between the z/OS AI Framework configuration workflow and Network Configuration Assistant.....	23
Upgrading the z/OS AI Framework configuration from AI System Services for IBM z/OS 1.1 to 1.2.....	24
Chapter 5. Using the AI Control Interface for IBM z/OS.....	27
AI Control Interface.....	27
Getting started in AI Control Interface.....	27
Terms you should know.....	28
AI use cases.....	28
Training a system.....	29
Managing intelligent operations.....	29
Chapter 6. Operations.....	35
Operating IBM Z Common Data Provider.....	35
Operating AI Base Component for z/OS.....	35

Starting AI Base Component for IBM z/OS (AIBPROC).....	35
Stopping AI Base Component for IBM z/OS (AIBPROC).....	36
Chapter 7. Performance considerations.....	37
Chapter 8. Service and support considerations.....	39
Chapter 9. Reference information for system programmers.....	41
MVS system messages.....	41
AIB messages.....	41
z/OSMF messages.....	58
IZU messages.....	58
MVS system codes.....	70
System completion codes.....	70
Diagnosis information.....	70
ENQ/DEQ summary.....	70
Component trace.....	71
SVC dump directory.....	74
Health checks.....	75
AI Base Component for IBM z/OS checks (IBMAIB).....	75
Part 3. AI providers and use cases.....	77
Chapter 10. AI-powered WLM batch initiator management.....	79
AI in WLM.....	80
Controlling and managing AI-powered WLM batch initiator management.....	82
Reference information related to the AI-powered WLM use case.....	84
PARMLIB.....	84
Event notification facility.....	84
SMF records.....	85
Monitor III measurement tables.....	87
IWM messages.....	89
Accessibility.....	93
Notices.....	95
Terms and conditions for product documentation.....	96
IBM Online Privacy Statement.....	97
Policy for unsupported hardware.....	97
Minimum supported hardware.....	97
Trademarks.....	98
Index.....	99

Figures

- 1. AI Framework for IBM z/OS components..... 5
- 2. z/OS batch job assignment by service classes..... 79
- 3. Workload and initiator behavior for batch workload management vs. AI-powered batch workload management..... 80
- 4. The AI Control Interface for IBM z/OS: Manage - Components view 83

Tables

1. z/OS AI Framework: Required products, FMIDs, and components.....	8
2. Default network port number usage.....	15
3. Columns in the Systems table.....	30
4. Persistent columns in the components table.....	31
5. Dependent columns in the components table.....	32
6. Targeted actions for the components table.....	32
7. Untargeted actions for the components table.....	32
8. Controls in the AI simulation chart.....	33
9. Fields in the component sidebar.....	33
10. Targeted actions.....	34
11. General actions.....	34
12. Summary of major and minor ENQ/DEQ names and resources.....	70
13. Summary of BCP component traces that use the component trace service.....	71
14. Determining if a component has a parmlib member.....	71
15. Component trace options.....	71
16. Trace request options for component traces.....	71
17. Location of trace buffers for components.....	71
18. How to request SVC dumps for component traces.....	71
19. Subcommands that format component trace records.....	71
20. ENF macro event codes.....	85

About Infusing AI into IBM z/OS

Purpose of this information

This content solution is a comprehensive collection of all the information necessary to understand and use the support that z/OS provides for infusing AI into various system management activities, the AI Framework for IBM z/OS. Some of this information also appears in individual publications elsewhere in the z/OS library or via other interfaces, such as the z/OS Management Facility (z/OSMF).

Who should read this information

This information is intended for systems programmers or other persons responsible for setting up and managing the AI Framework for z/OS and the providers of AI-enhanced use cases.

About this information

This comprehensive content collection consists of original content, as well as relevant content excerpted from the following z/OS publications:

- [*IBM Health Checker for z/OS User's Guide*](#)
- [*z/OS Data Gatherer Programmer's Guide*](#)
- [*z/OS Introduction and Release Guide*](#)
- [*z/OS MVS Diagnosis: Reference*](#)
- [*z/OS MVS Diagnosis: Tools and Service Aids*](#)
- [*z/OS MVS Initialization and Tuning Reference*](#)
- [*z/OS MVS Planning: Workload Management*](#)
- [*z/OS MVS Programming: Authorized Assembler Services Guide*](#)
- [*z/OS MVS System Codes*](#)
- [*z/OS MVS System Commands*](#)
- [*z/OS MVS System Management Facilities \(SMF\)*](#)
- [*z/OS MVS System Messages, Vol 1 \(ABA-AOM\)*](#)
- [*z/OS MVS System Messages, Vol 9 \(IGF-IWM\)*](#)

This comprehensive content collection also includes relevant content excerpted from the following z/OSMF online help information:

- [*AI Control Interface*](#)
- [*AI Control Interface Messages*](#)
- [*z/OSMF Workflows task*](#)

All of these information resources are available in [IBM Documentation for z/OS \(www.ibm.com/docs/en/zos\)](http://www.ibm.com/docs/en/zos).

How to provide feedback to IBM

We welcome any feedback that you have, including comments on the clarity, accuracy, or completeness of the information. For more information, see [How to send feedback to IBM](#).

Summary of changes

This information includes terminology, maintenance, and editorial changes. Technical changes or additions to the text and illustrations for the current edition are indicated by a vertical line to the left of the change.

Note: IBM z/OS policy for the integration of service information into the z/OS product documentation library is documented on the z/OS Internet Library under [IBM z/OS Product Documentation Update Policy](http://www.ibm.com/docs/en/zos/latest?topic=zos-product-documentation-update-policy) (www.ibm.com/docs/en/zos/latest?topic=zos-product-documentation-update-policy).

Summary of changes for z/OS 3.2

This information contains no technical changes for this release.

Summary of changes for z/OS 3.1

The following content is new or changed in z/OS 3.1.

New

The following content is new.

June 2025 refresh

- Information about AI System Services for IBM z/OS 1.2 and Machine Learning for IBM z/OS 3.2 Core Edition is added, and the topic, “[Upgrading the z/OS AI Framework configuration from AI System Services for IBM z/OS 1.1 to 1.2](#)” on page 24, is added. (APAR OA67865)
- The topic, “[Importing and exporting configuration variables in a property file](#)” on page 22, is added. (APAR OA67551)

April 2025 refresh

- Message AIB0019I is added in “[AIB messages](#)” on page 41. (APAR OA66203)

August 2024 refresh

- Information is added about “[Avoiding AT-TLS configuration conflicts between the z/OS AI Framework configuration workflow and Network Configuration Assistant](#)” on page 23.

June 2024 refresh

- Information is added about reviewing AI simulation data in the z/OSMF AI Control Interface task; see [Chapter 5, “Using the AI Control Interface for IBM z/OS,”](#) on page 27. This enhancement introduces data visualization to compare the number of active batch initiators managed by WLM to the number of initiators predicted by AI in simulation mode. (APAR PH56709)
- Messages IZUAM0105E–IZUAM0116W are added in “[IZU messages](#)” on page 58. (APAR PH56709)

December 2023 refresh

- Links to the white paper, *IBM z/OS 3.1 AI-Powered WLM Batch Initiator Management Resource and Tuning Guidelines*, are added in “[Hardware and software requirements](#)” on page 13 and [Chapter 7, “Performance considerations,”](#) on page 37.

October 2023 refresh

- [Chapter 8, “Service and support considerations,”](#) on page 39 is added.

Changed

The following content is changed.

July 2025 refresh

- IBM Z® Common Data Provider (ZCDP) 5.1 at PTF level UJ96789/UJ97146 or later requires IBM Semeru Runtime® Certified Edition for z/OS, Version 17 (5655-UA1). See [“Software requirements” on page 14](#).

June 2025 refresh

- The z/OS AI Framework configuration workflows are updated. Use of exporting variable input values to a property file has been expanded to simplify the workflows and eliminate redundant prompting, and the WLM use case configuration workflow has been separated from the main AI framework configuration workflow. (APARs OA67551, OA67728)

Updates are made in:

- [“Using the AI Framework for IBM z/OS Configuration Workflow” on page 17](#)
- [“Adding another z/OS system to your z/OS AI Framework configuration” on page 20](#)
- The topics, “AI in WLM” and “Configuring and managing the AI-powered WLM batch initiator management use case,” in Part 3, [“AI providers and use cases,” on page 77](#).
- The figure showing the AI Control Interface for z/OS is updated in “Using the AI Control Interface for z/OS to manage AI-powered WLM batch initiator management” in Part 3, [“AI providers and use cases,” on page 77](#).

May 2025 refresh

- Updates are made in [“Using the AI Framework for IBM z/OS Configuration Workflow” on page 17](#).

December 2024 refresh

- Additional guidance about using the correct product levels and using the AI Framework for IBM z/OS Configuration Workflow to configure IBM Z Common Data Provider (ZCDP) and Machine Learning for IBM z/OS Core Edition (MLz Core) is provided in Chapter 2, [“z/OS AI Framework software and product packaging,” on page 7](#) and [“Using the AI Framework for IBM z/OS Configuration Workflow” on page 17](#).

October 2024 refresh

- The table, [Table 1 on page 8](#), is reformatted for better presentation and clarity.

June 2024 refresh

- Information about IBM Semeru Version 11 is updated in [“Installing the software” on page 15](#) and [“Using the AI Framework for IBM z/OS Configuration Workflow” on page 17](#).

May 2024 refresh

- The product name, Watson Machine Learning for IBM z/OS (WMLz), is updated to Machine Learning for IBM z/OS (MLz).
- Information about IBM Semeru Version 11 is updated in [“Software requirements” on page 14](#).

December 2023 refresh

- The hardware requirements are updated in [“Hardware requirements” on page 13](#).

October 2023 refresh

- None.

Part 1. Overview of the AI Framework for IBM z/OS

This part presents an overview and introduction to the AI Framework for IBM z/OS (z/OS AI Framework), including useful terminology, descriptions of the z/OS AI Framework components, and various packaging and delivery options for the z/OS AI Framework software.

Chapter 1. Introduction to the AI Framework for z/OS

IBM z/OS 3.1 introduces artificial intelligence (AI) that is infused into the operating system to provide intelligent automation and accelerated inferencing at scale to extract and leverage valuable data insights. The AI capability in z/OS consists of the following deliverables:

AI Framework for IBM z/OS

AI Framework for IBM z/OS (z/OS AI Framework) supports augmenting z/OS with intelligence that helps to optimize IT processes, simplify management, reduce skill requirements, and improve performance. AI Framework for z/OS includes components that provide data collection, AI model training, deployment, and monitoring, and AI inferencing and scoring. Further, AI Framework for z/OS provides a component that allows other z/OS components to plug into the framework by enabling traditional z/OS programming interfaces, such as assembler services, to access REST API functions for simpler AI deployment.

AI System Services for IBM z/OS

AI System Services for IBM z/OS (z/OS AI Services; also referred to as "AISS" in this documentation) delivers foundational AI capabilities and represents one of the key components of the z/OS AI Framework. AI System Services for z/OS provides the following capabilities:

- IT data ingestion and filtering capabilities that enable the collection of data for model training and inferencing purposes
- AI model server capabilities that support the AI model lifecycle, including AI model training, deployment, inferencing, monitoring, and retraining

The integration of AI System Services for z/OS with the rest of the z/OS AI Framework components enable prebuilt AI models to be put into operation to simplify the management of z/OS and its offerings by augmenting them with AI, reducing skill requirements, optimizing IT processes, and improving performance.

AI-powered WLM batch initiator management

Workload management (WLM) is the first z/OS component to plug into the z/OS AI Framework as a provider of an AI-infused use case. AI-powered WLM batch initiator management is designed to intelligently predict upcoming batch workload and proactively optimize system resources to meet the demand.

The AI Framework for z/OS, including AI System Services for z/OS, offers an AI platform that seamlessly integrates a set of components to enable the operation and use of AI-infused capabilities by z/OS base components, starting with the initial use case for AI-powered WLM batch initiator management. The platform is intended to offer simplified installation, setup, and management of AI-infused capabilities without requiring additional data science or AI skills. The platform is designed to pave the way for AI use case providers to harness the foundational AI capabilities and accelerate the deployment of future AI use cases.

Useful terminology

Using the AI-infused capabilities provided by the AI Framework for IBM z/OS does not require you to have any special skills in AI or data science. Becoming familiar with the following terminology will aid in your understanding of the concepts discussed in this documentation.

artificial intelligence (AI)

The emulation of natural intelligence by a machine.

data science

An interdisciplinary field that uses algorithms, procedures, and processes to examine large amounts of data in order to uncover hidden patterns, generate insights, and direct decision making. To create prediction models, data scientists use advanced machine learning algorithms to sort through, organize, and learn from structured and unstructured data. Data science often starts with an initial

investigation of business problems to identify potential solutions making use of artificial intelligence and machine learning.

deployment

In a model server, the process of making a trained model available for inferencing and scoring requests.

inferencing

The process of using a trained machine learning model to make predictions on input data.

machine learning (ML)

A method to learn and adapt without the need to provide explicit instructions, using statistics and algorithms to analyze and draw inferences from patterns in data.

model

A set of functions and algorithms that have been trained and tested on a body of data to provide predictions or decisions. Before it can be used, a trained model must be deployed in a model server.

model server

A z/OS software product that hosts and manages the training applications and models, including training, versioning, deployment, and monitoring.

provider

A z/OS component or product that plugs into the AI Framework for IBM z/OS for easier exploitation of AI capabilities for a particular use case.

scoring

The process of assigning a numerical value (score) to a particular entity or outcome, for instance, to determine the likelihood of certain outcomes.

training

In a model server, the process of preparing a model for productive use by using a training application to analyze and learn from a body of historical operational data. Periodically retraining a model as the body of historical data grows and changes over time helps ensure that the model continues to perform optimally.

training application

The software used to train a model.

use case

In z/OS, a practical application, such as a particular system management function, that a provider has enhanced with AI-driven insights.

workflow

A guided set of sequential steps or actions to perform a particular task, such as configuring a component or product. In z/OS, workflows are run in z/OSMF and often replace traditional written instructions. Workflow steps can consist of a mix of manual or automated actions. Steps can be assigned to different people who are responsible for their completion, and progress can be tracked.

z/OS AI Framework components

The AI Framework for IBM z/OS consists of 5 major component areas that, together, enable AI-enhanced capabilities within the z/OS system.

Figure 1 on page 5 shows the major components that form the AI Framework for IBM z/OS.

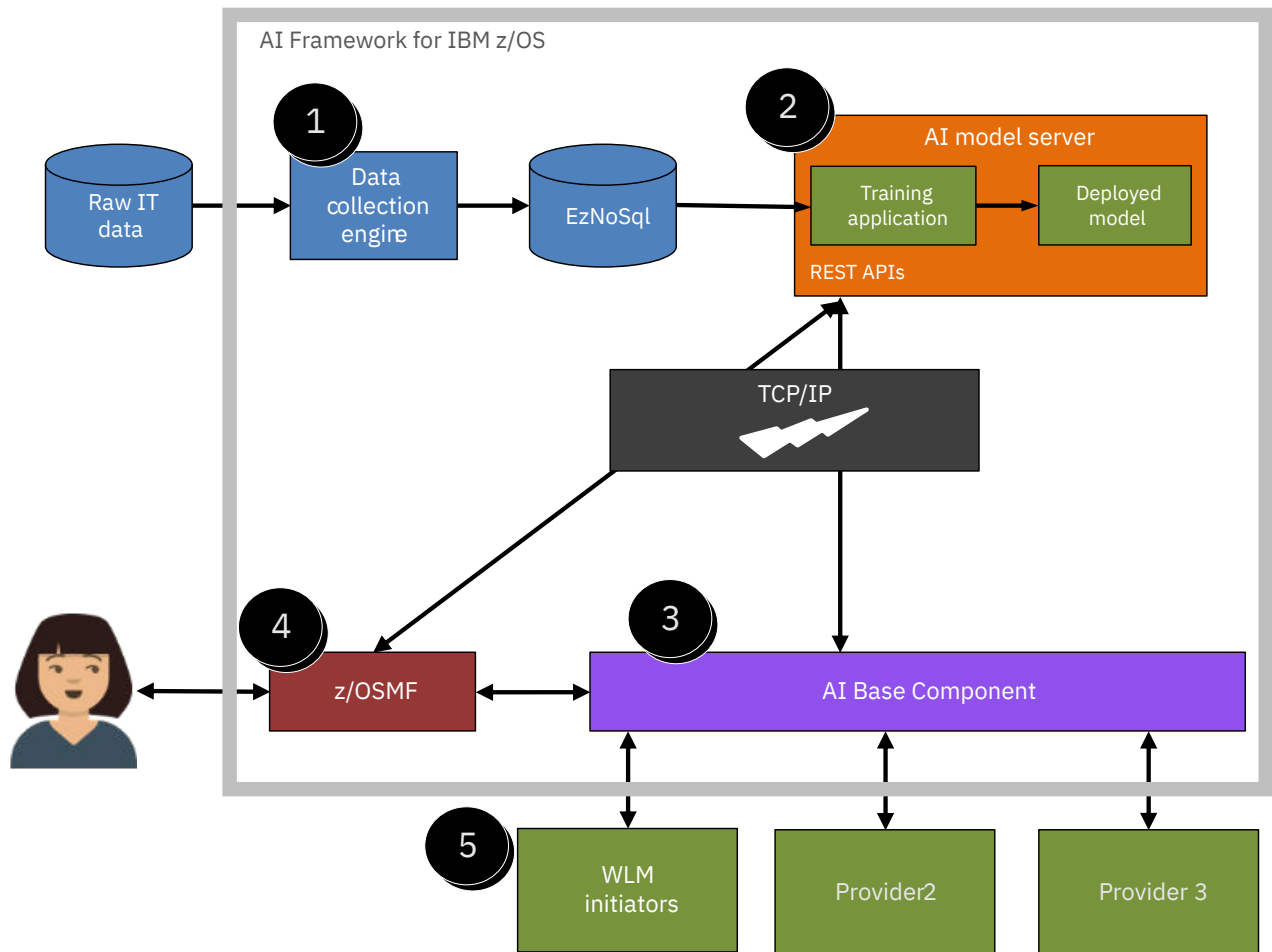


Figure 1. AI Framework for IBM z/OS components

The following component descriptions correspond to the numbered components shown in [Figure 1](#) on page 5:

1 – Data collection

Data collection provides a common way to collect IT data for use in AI model training.

The *data collection engine* performs the following functions:

- Collects and parses the raw IT data.
- Streams the data to the data store.

AI Framework for IBM z/OS uses IBM Z Common Data Provider (ZCDP) as the data collection engine.

EzNoSQL for z/OS, a component of DFSMS, serves as the data store to hold data for training the AI models.

- Implemented using VSAM data sets.
- Accessed by the model training pipeline and, optionally, by the deployed model.
- Can be used both for training and for the deployed model to store its own data.

2 – AI model server

The AI model server hosts and manages the AI models.

- Manages model training, versioning, deploying, and monitoring.
- Supports failover for high availability.
- Accessed by the system via REST APIs.

AI Framework for IBM z/OS uses Machine Learning for IBM z/OS Core Edition (MLz Core) as the AI model server.

3 — AI Base Component

AI Base Component for IBM z/OS (z/OS AI Base), a new component of the BCP element, assists use-case providers in communicating with the AI model server. This support enables traditional z/OS components to use AI models without the need for frequent updates to the latest technologies.

- Access is via traditional z/OS assembler services (macros).
- Handles connection to REST APIs.

4 — User interface (z/OSMF)

The user interface is the primary means by which end users work with the z/OS AI Framework and manage models, and is provided through IBM z/OS Management Facility (z/OSMF).

- A z/OSMF *workflow*, known as the AI Framework for IBM z/OS Configuration Workflow, guides you through the installation process with detailed configuration steps for each of the framework components.
- The AI Control Interface for IBM z/OS, a new task on the z/OSMF desktop, provides AI model management. You use the z/OS AI Interface to initiate training of a model for a use case and to enable or disable the AI mode for that use case or place it into simulation mode.

5 — AI providers and use cases

Providers can plug into the AI framework to perform the following functions for simpler deployment of AI-driven use cases:

- Define data collection.
- Create the model training pipeline.
- Call the z/OS AI Base component.
- Extend the user interface.

Workload management (WLM) is the first provider of an AI use case, AI-powered WLM batch initiator management, that uses the z/OS AI Framework. The use case proactively starts WLM-managed initiators based on predictive insights on upcoming batch workloads.

The framework is designed to be expandable to include additional providers and use cases.

Chapter 2. z/OS AI Framework software and product packaging

AI Framework for IBM z/OS (z/OS AI Framework) requires the following software products to provide certain framework functionality. You must order and install these products at the specified levels in addition to the IBM z/OS 3.1 base product.

Important: You must use the specified FMIDs of the following products. If you have earlier or later levels of these products installed on your system, you cannot use them with the z/OS AI Framework.

IBM Z Common Data Provider 5.1

The data collection function of the z/OS AI Framework uses IBM Z Common Data Provider (ZCDP) as the data collection engine. ZCDP provides the infrastructure for accessing IT operational data (such as SMF records) from z/OS systems and streaming it to the analytics platform in a consumable format.

The ZCDP 5.1 FMID (HHBO510) is typically packaged and installed as part of other IBM Z priced products; however, these other products are not required for you to use the z/OS AI Framework. If you do not already use any other such products that include the level of ZCDP that the z/OS AI Framework requires, you can obtain and install the correct level of ZCDP as part of AI System Services for IBM z/OS, as described in [“Preferred ordering and packaging scenario” on page 7](#).

Machine Learning for IBM z/OS Core Edition 3.1 or 3.2

The AI model server function of the z/OS AI Framework uses Machine Learning for IBM z/OS Core Edition (MLz Core), which is a machine learning solution that runs on the IBM Z platform. (Machine Learning for IBM z/OS (MLz) was formerly known as Watson Machine Learning for IBM z/OS (WMLz).)

The MLz Core FMID (HAQN310 for MLz Core 3.1 or HAQN320 for MLz Core 3.2) is included with the priced product, Machine Learning for IBM z/OS Enterprise Edition (MLz Enterprise); however, MLz Enterprise is not required for you to use the z/OS AI Framework. If you do not already use a level of MLz Enterprise that includes the level of MLz Core that the z/OS AI Framework requires, you can obtain the correct level of MLz Core as part of AI System Services for IBM z/OS, as described in [“Preferred ordering and packaging scenario” on page 7](#).

All other functional components of the z/OS AI Framework (as shown in [Figure 1 on page 5](#)) are delivered as part of the z/OS 3.1 base product.

Preferred ordering and packaging scenario

To simplify ordering and installation, IBM offers AI System Services for IBM z/OS (5655-164), which packages the ZCDP and MLz Core FMIDs together under a single, *zero-charge* product ID (PID).

- AI System Services for IBM z/OS 1.1 (AISS 1.1) contains the ZCDP 5.1 and MLz Core 3.1 FMIDs, and requires Java 11, as described in [“Software requirements” on page 14](#).
- AI System Services for IBM z/OS 1.2 (AISS 1.2) contains the ZCDP 5.1 and MLz Core 3.2 FMIDs, and requires Java 17, as described in [“Software requirements” on page 14](#). This is the preferred option, as MLz Core 3.2 offers additional functionality and security. AISS 1.2 requires APAR OA67865 on z/OS 3.1.

If you are migrating from z/OS 2.4 or 2.5 to z/OS 3.1 and you do not already have any other products that include the ZCDP and MLz Core FMIDs, you can order the AI System Services for IBM z/OS PID along with the z/OS 3.1 PID (5655-ZOS) and receive both products in one portable software instance zone.

Tip: Installing both z/OS and AISS in the same zone makes applying future software maintenance easier than if the products are in different zones.

Table 1 on page 8 shows the PIDs and the FMIDs and components from each product that the z/OS AI Framework requires.

Table 1. z/OS AI Framework: Required products, FMIDs, and components

This product...	Contains these FMIDs...	Which include these components...
Option 1: AI System Services for IBM z/OS 1.1 PID 5655-164 (\$0)	ZCDP 5.1 FMID HHBO510	–
	MLz Core 3.1 FMID HAQN310	–
Option 2 (preferred): AI System Services for IBM z/OS 1.2 PID 5655-164 (\$0)	ZCDP 5.1 FMID HHBO510	–
	MLz Core 3.2 FMID HAQN320	–
IBM z/OS 3.1 PID 5655-ZOS	BCP FMID HBB77E0	<ul style="list-style-type: none"> • AI Base Component for z/OS • WLM (AI use case provider)
	z/OSMF FMID HSMA310	<ul style="list-style-type: none"> • AI Control Interface for z/OS
	DFSMS FMID HDZ3310	<ul style="list-style-type: none"> • VSAM RLS • EzNoSQL for z/OS

A benefit of this packaging scenario is that, in addition to receiving ZCDP and MLz Core at no charge, you are assured that ZCDP and MLz Core are at the proper levels that the z/OS AI Framework requires. If you are using a previous installation of ZCDP or MLz Core or both, be sure they are at the correct levels.

Note: You can use this same packaging strategy if you already have z/OS 3.1 and you later want to add AI System Services for z/OS so that you can use the z/OS AI Framework, and you do not already have any other products that include ZCDP and MLz Core. You can order AI System Services for z/OS and receive ZCDP and MLz Core in one portable software instance zone.

Other ordering and packaging scenarios

The following scenarios describe various combinations of ordering and packaging for z/OS, AI System Services for z/OS, MLz Enterprise, and ZCDP.

• You already have z/OS 3.1 and MLz Enterprise, but no ZCDP.

If you already have z/OS 3.1, and you already have MLz Enterprise 3.1 or 3.2 (which includes the MLz Core FMID (HAQN310 or HAQN320)), but you need the ZCDP FMID (HHBO510), you have 2 options:

- You can order and install another IBM product that includes the ZCDP FMID, if you already intend to use that product.
- You can order and install the zero-charge AI System Services for z/OS, but this will cause duplication of the MLz Core FMID, which you already have as part of MLz Enterprise. You have 3 options to *manually* mitigate the duplication:
 - Carefully merge the two instances.
 - Carefully delete the MLz Core FMID from the portable software instance for your AISS order.
 - Do not use a z/OSMF portable software instance; instead, use CBPDO but do not install the MLz Core FMID that comes with AISS.

Note: If you order AI System Services for z/OS *at the same time* as another product that also contains ZCDP, they will install into the same zone and share the FMID for ZCDP.

• You already have z/OS 3.1 and ZCDP, but no MLz.

If you already have z/OS 3.1, and you already have the ZCDP FMID (HHBO510) as part of another IBM product, but you need the MLz Core FMID (HAQN310 or HAQN320), you have 2 options:

- You can order and install MLz Enterprise, which will also install the MLz Core FMID that you need for the z/OS AI Framework. This approach makes sense if you already intend to use MLz Enterprise for other purposes apart from the z/OS AI Framework.
- You can order and install the zero-charge AI System Services for z/OS, but this will cause duplication of the ZCDP FMID, which you already have as part of another product. You have 3 options to *manually* mitigate the duplication:
 - Carefully merge the two instances.
 - Carefully delete the ZCDP FMID from the portable software instance for your AISS order.
 - Do not use a z/OSMF portable software instance; instead, use CBPDO but do not install the ZCDP FMID that comes with AISS.

Note: If you order AI System Services for z/OS *at the same time* as MLz Enterprise, they will install into the same zone and share the FMID for MLz Core.

- **You already have z/OS 3.1 and ZCDP and MLz Enterprise.**

If you already have z/OS 3.1 and MLz Enterprise (which includes the MLz Core FMID (HAQN310 or HAQN320)), and you also already have the ZCDP FMID (HHBO510) as part of another IBM product, you do not need to order or install anything else; you already have all the necessary FMIDs on your system.

ZCDP and MLz Core instances are not shared

Regardless of the particular ordering, packaging, and installation process you use, the z/OS AI Framework configures its own running instances of ZCDP and MLz Core, including the Spark runtime environment that MLz Core uses. These separately configured instances are not shared with any other products, and any existing ZCDP and MLz Core configurations remain unaffected. This approach ensures that these separate instances are properly configured for use with the z/OS AI Framework (for instance, proper security setup).

Important: The AI Framework for IBM z/OS Configuration Workflow (described later) prompts you to specify the locations where the correct levels of ZCDP and MLz Core are installed on your system and configures new, separate instances of these products that are specifically for use with the z/OS AI Framework.

You *must* use the AI Framework Configuration Workflow to configure these new instances of ZCDP and MLz Core for use with the z/OS AI Framework. Do *not* attempt to use separate configuration mechanisms (such as other z/OSMF plug-ins or GUIs that are provided by those products) to configure new instances, nor share existing instances that are used with other products; otherwise, the z/OS AI Framework will not work correctly. Use the AI Framework Configuration Workflow instead, as described later in [Chapter 4, “Configuring the z/OS AI Framework,”](#) on page 17.

Part 2. Getting started with the AI Framework for IBM z/OS

This part presents information to help you get started with the AI Framework for IBM z/OS, including:

- Planning and installation
- Post-installation configuration using z/OSMF workflows
- Using the AI Control Interface for IBM z/OS
- Steady-state operations
- Reference information, such as message explanations and diagnosis information

Chapter 3. Planning and installation

Ensure that your system meets the hardware and software requirements for AI Framework for IBM z/OS before you attempt to install and configure the z/OS AI Framework software.

Hardware and software requirements

AI Framework for IBM z/OS requires the following hardware and software.

Tip: You can find additional practical guidance about these hardware and software requirements from IBM performance analysts in the white paper, [IBM z/OS 3.1 AI-Powered WLM Batch Initiator Management Resource and Tuning Guidelines \(www.ibm.com/downloads/cas/A6LG1M38\)](https://www.ibm.com/downloads/cas/A6LG1M38).

Hardware requirements

The z/OS AI Framework requires the following hardware:

- Any of the following IBM Z servers:
 - IBM z17 — all models, including ME1 (machine type 9175)
 - IBM z16 — all models, including A01 (machine type 3931) and A02 (machine type 3932)
 - IBM z15 — all models, including T01 (machine type 8561) and T02 (machine type 8562)
- Coupling facility (internal or external; required by EzNoSQL/VSAM record level sharing (RLS))
 - You will need a coupling facility cache structure of the appropriate size:
 - If you specify DIRONLY for the SMS DATACLAS RLS CF cache value (not caching data in the cache structure), a smaller structure size should suffice.
 - If you do not specify DIRONLY (caching data in the cache structure), a larger structure size will be needed. Use the [Coupling Facility Structure Sizer \(CFSizer\)](#) tool for guidance on sizing the structure.

Guideline: The cache size (and local buffer pool size) should be a minimum of 20% of the data size.

- If possible, do not share the cache structure. The guideline is to have separate caches for different RLS exploiters. For instance, if you are using HSM, it should have its own cache and the EzNoSQL databases should have their own. (It is possible to allow both HSM and EzNoSQL to share a cache, but you will need to increase the structure size accordingly.)
- The configuration workflow (discussed later in [“Using the AI Framework for IBM z/OS Configuration Workflow”](#) on page 17) will guide you in configuring EzNoSQL. However, if you are interested in learning more detailed information, see [Record-level Sharing CF Caching in z/OS DFSMS Using Data Sets and EzNoSQL for z/OS documentation \(github.com/IBM/IBM-Z-zOS/blob/main/zOS-EzNoSQL/EzNoSQL%20Documentation.md\)](#).
- IBM Z Common Data Provider (ZCDP) runs on each z/OS LPAR where SMF records are to be collected for the z/OS AI Framework and requires the following minimum system capacity on each LPAR:
 - Memory:** 1 GB
 - DASD:** 8 GB
 - zIIP:** ZCDP can offload work to IBM Z Integrated Information Processors (zIIPs) if one or more zIIPs are online before the ZCDP components are started.
- Machine Learning for IBM z/OS Core Edition (MLz Core) runs on the z/OS LPAR where the z/OSMF server (AI Control Interface task) runs and requires the following minimum system capacity:
 - Processor:** 1 shared zIIP or shared CP (2 suggested)
 - Memory:** 31 GB
 - DASD:** 10 GB (50 GB suggested)

These capacities are in addition to those required for ZCDP running on the same LPAR.

Note: The preceding requirements are based on IBM internal testing and individual product documentation. Actual processor, memory, and DASD requirements will vary based on the number of participating z/OS LPARs, and the number of service classes per LPAR, the workload, and contention for shared resources. IBM strongly suggests that you monitor your system to ensure you have adequate memory to avoid paging, and adequate processor resource to avoid processor delays, both of which can adversely affect performance.

Software requirements

The z/OS AI Framework requires the following software:

- **IBM z/OS 3.1 or later**

The z/OS AI Framework requires the following z/OS base elements:

- Base Control Program (BCP), which includes WLM and AI Base Component for z/OS
- DFSMS, which includes EzNoSQL for z/OS (Learn more about [EzNoSQL for z/OS \(www.ibm.com/support/z-content-solutions/eznosql\)](http://www.ibm.com/support/z-content-solutions/eznosql).)
 - EzNoSQL for z/OS uses VSAM record level sharing (RLS), which requires a coupling facility. Further, the VSAM server address space (SMSVSAM) must be initialized on each z/OS system in the sysplex that will be part of the z/OS AI Framework configuration.

EzNoSQL requires the availability of one or more SMS storage classes (STORCLAS) which contain an RLS CACHESET with the name of a valid coupling facility cache structure. The STORCLAS names can be provided to the application architect or assigned by the SMS ACS routines.
 - The EzNoSQL database requires enough space to accommodate up to 60 days of SMF data. To meet this requirement, IBM estimates that 2000 cylinders of primary space and 500 cylinders of secondary space would be needed.
 - Also see the coupling facility requirements related to the RLS cache structure earlier in “[Hardware requirements](#)” on page 13.
 - EzNoSQL must be available on each system that will be part of the z/OS AI Framework where SMF records will be collected.
- z/OS Management Facility (z/OSMF), which includes the AI Control Interface for z/OS
 - The z/OSMF data directory must be mounted on a shared, global path, /global/zosmf, as described in Step 2: Run job IZUMKFS to create the z/OSMF user file system in *IBM z/OS Management Facility Configuration Guide*. This allows z/OS AI Framework components that are running on remote systems to access z/OSMF persistent data.
- z/OS UNIX System Services

- **IBM Z Common Data Provider (ZCDP) 5.1**

- The z/OS AI Framework requires its own running instance of ZCDP on each system that will be part of the z/OS AI Framework where SMF records will be collected.

- **Machine Learning for IBM z/OS Core Edition (MLz Core) 3.1 or 3.2**

- MLz Core also requires IBM WebSphere Application Server for z/OS Liberty, version 22.0.0.9 or later.
- MLz Core includes the Spark runtime environment, which will be configured as part of the overall z/OS AI Framework configuration process (described later).
- The home directory for MLz Core must have at least 10 GB of available space (50 GB suggested).
- On z/OS 3.1, APAR OA67865 is required if you wish to use MLz Core 3.2.

- A **Java™ runtime**, as follows:

- MLz Core 3.1 and ZCDP 5.1 prior to PTF level UJ96789/UJ97146 require **IBM Semeru Runtime Certified Edition for z/OS, Version 11 (5655-DGJ) (SR15 or later)**

Note: As of April 2024, z/OS 3.1 ships with IBM Semeru Runtime Certified Edition for z/OS, Version 17. However, you can still download and install IBM Semeru 11, which is required when using MLz Core 3.1 with the z/OS AI Framework. You must install IBM Semeru 11 in addition to—not instead of—IBM Semeru 17. You can obtain IBM Semeru 11 from Java SDK Products on z/OS (www.ibm.com/support/pages/java-sdk-products-zos#v11).

- MLz Core 3.2 and ZCDP 5.1 at PTF level UJ96789/UJ97146 or later require **IBM Semeru Runtime Certified Edition for z/OS, Version 17 (5655-UA1)**

This is not an exhaustive list. The list covers the major software items that might require advance planning. Other software setup requirements, such as access and authorization to or for various components, are identified during the configuration phase using the AI Framework for z/OS Configuration Workflow. You can browse through the configuration workflow before you actually start performing the steps to learn what the configuration process entails. (See [Chapter 4, “Configuring the z/OS AI Framework,”](#) on page 17.)

Default network port usage

AI Framework for IBM z/OS uses several network ports for communication between various framework components and services.

Table 2 on page 15 lists the default network port numbers that will be defined during the configuration phase. The AI Framework Configuration Workflow (described in [“Using the AI Framework for IBM z/OS Configuration Workflow”](#) on page 17) allows you to override these defaults.

Table 2. Default network port number usage

Service	AI Framework Configuration Workflow variable name	Default port number
ZCDP Data Streamer	ZCDP_DS_Port	51401
MLz core services	WMLz_Core_Services_Port	11442
MLz scoring service	WMLz_Scoring_Services_Port	8388
z/OS Spark master	WMLz_Spark_Master_Port	7077
z/OS Spark master REST API	WMLz_Spark_Master_Api_Port	6066
z/OS Spark master UI	WMLz_Spark_Webui_Port	8080
z/OS Spark worker	WMLz_Spark_Worker_Port	8088
z/OS Spark worker UI	WMLz_Spark_Worker_Webui_Port	8081
z/OS Spark driver	WMLz_Spark_Driver_Port	6060
z/OS Spark block manager	WMLz_Spark_Block_Manager_Port	6061
z/OS Spark driver block manager	WMLz_Spark_Driver_Block_Manager_Port	9090

Installing the software

Complete this task to install all software products that are necessary for the AI Framework for IBM z/OS.

Before you begin

Be sure your system meets the requirements described in [“Hardware and software requirements”](#) on page 13.

About this task

Detailed instructions for installing z/OS software products are provided in a *program directory*, which accompanies your software order. You might need to use more than one program directory if you are ordering and installing multiple z/OS products.

Procedure

1. Follow your normal IT procedures to install z/OS 3.1 (5655-ZOS) according to the instructions in *Program Directory for IBM z/OS 3.1* (www.ibm.com/docs/en/SSLTBW_3.1.0/pdf/e0zpdz00.pdf).
2. If you intend to use MLz Core 3.1 and your installation of z/OS 3.1 contains IBM Semeru Runtime Certified Edition for z/OS Version 17 instead of Version 11 (as explained earlier in “Software requirements” on page 14), download and install IBM Semeru 11 alongside IBM Semeru 17. (Do not remove IBM Semeru 17.)
3. Install the AI System Services for IBM z/OS (AISS) 1.1 or 1.2 (5655-164) software in either of the following ways:
 - If you *are* installing AISS 1.1 or 1.2 to obtain IBM Z Common Data Provider (ZCDP, FMID HHBO510) and Machine Learning for IBM z/OS Core Edition (FMID HAQN310 for MLz Core 3.1 or HAQN320 for MLz Core 3.2), follow your normal IT procedures to install the software according to the instructions in *Program Directory for AI System Services for IBM z/OS 1.1* (www.ibm.com/docs/en/SSLTBW_3.1.0/pdf/GI135664.pdf), GI13-5664-00, or *Program Directory for AI System Services for IBM z/OS 1.2* (www.ibm.com/docs/en/SSLTBW_3.1.0/pdf/GI13566401.pdf), GI13-5664-01, as appropriate.
 - If you *are not* installing AISS because you already have (or will have) the necessary ZCDP (HHBO510) and MLz Core (HAQN310 or HAQN320) FMIDs installed as part of other IBM products (as discussed in Chapter 2, “z/OS AI Framework software and product packaging,” on page 7), ensure that those products are successfully installed and are at the appropriate service (PTF) level. Consult *Program Directory for AI System Services for IBM z/OS 1.1* (www.ibm.com/docs/en/SSLTBW_3.1.0/pdf/GI135664.pdf), GI13-5664-00, or *Program Directory for AI System Services for IBM z/OS 1.2* (www.ibm.com/docs/en/SSLTBW_3.1.0/pdf/GI13566401.pdf), GI13-5664-01, as appropriate, to ensure that your installation meets the requirements.

Notes about this step: Be aware of the following points when installing the software:

- On z/OS 3.1, APAR OA67865 is required if you wish to use MLz Core 3.2.
- ZCDP and MLz Core must be installed on the same z/OS system where z/OSMF runs to configure the z/OS AI Framework.
- ZCDP must also be installed on each additional z/OS system that you intend to configure for use with the z/OS AI Framework.

Note that the z/OS AI Framework requires its own running instance of ZCDP to be configured on each participating system, which will happen later during the configuration phase. This is necessary because the ZCDP data streamer will be configured to process specific SMF record data and stream it to the data store (EzNoSQL).

Results

Upon completion, you should have z/OS 3.1 installed on your system (and, if necessary, IBM Semeru 11), along with the FMIDs for ZCDP and MLz Core.

What to do next

After successfully installing the software, continue with [Chapter 4, “Configuring the z/OS AI Framework,” on page 17.](#)

Chapter 4. Configuring the z/OS AI Framework

AI Framework for IBM z/OS (z/OS AI Framework) provides a z/OSMF-based workflow to guide you through the steps to configure the framework components. You use workflows from the **Workflows** task on the z/OSMF desktop.

What is a workflow?

If you are unfamiliar with using workflows in z/OSMF, see the following information to learn the basics.

Note: This information, along with complete information about z/OSMF workflows, is available as online help within z/OSMF and in [z/OSMF Workflows task \(www.ibm.com/docs/en/zos/latest?topic=help-workflows\)](http://www.ibm.com/docs/en/zos/latest?topic=help-workflows) in IBM Documentation.

Workflows overview

The z/OSMF Workflows task can help you guide the activities of system programmers, security administrators, and others at your installation who are responsible for managing the configuration of the system. The Workflows task provides a framework for these activities in the form of structured procedures that are known as *workflows*.

Using the Workflows task, you can assign individual work items (the workflow "steps") to performers and track their progress. Depending on how the workflow is designed, the Workflows task can offer wizards to assist with creating UNIX files and z/OS data sets, and submitting work to run on z/OS.

Using the AI Framework for IBM z/OS Configuration Workflow

Complete this task to learn how to use the AI Framework for IBM z/OS Configuration Workflow to configure the z/OS AI Framework for the first time. The actual configuration steps and actions are described in detail within the workflow itself.

Even if you are already using certain products, such as IBM Z Common Data Provider (ZCDP) or Machine Learning for IBM z/OS Enterprise Edition (which includes MLz Core), at the required levels, you must use *this* workflow to configure separate running instances of those products that are specifically for use with the z/OS AI Framework.



Attention: Do *not* attempt to use the configuration mechanisms provided by individual products (such as ZCDP or MLz) to configure those products for use with the z/OS AI Framework, and do *not* attempt to use or share existing instances of those products that were previously configured for other purposes, as doing so will not work correctly with the z/OS AI Framework.

Before you begin

Before you run the workflow to configure the z/OS AI Framework, complete the following manual steps:

- Be sure your system meets the requirements described in [“Hardware and software requirements”](#) on page 13 and that you have installed all the necessary software, as described in [“Installing the software”](#) on page 15.

Note: You must specify the location (directory path) of IBM Semeru Runtime Certified Edition for z/OS (which is required to run z/OS AI Framework) as the **Java installation path** during step 2, "Collect input for all components to configure the AI Framework for IBM z/OS," of the workflow.

- Set **Honor Priority** to NO for the service class for MLz and EzNoSQL, as described in [“Avoiding zIIP-eligible work executing on general purpose CPs”](#) on page 37.
- Ensure the availability of one or more SMS storage classes (STORCLAS) which contain an RLS CACHESET with the name of a valid CF cache structure. The STORCLAS names can be provided to the application architect or assigned by the SMS ACS routines.

- Ensure that the VSAM server address space (SMSVSAM) is initialized on each z/OS system in the sysplex that will be part of the z/OS AI Framework configuration.
- If you use the z/OSMF Network Configuration Assistant (NCA), see [“Avoiding AT-TLS configuration conflicts between the z/OS AI Framework configuration workflow and Network Configuration Assistant” on page 23.](#)

About this task

The AI Framework for IBM z/OS Configuration Workflow guides you, step-by-step, through the process to configure the z/OS AI Framework, on a z/OS system for the first time.

Unlike traditional procedural documentation where the instructions and the actions you perform are separate, a workflow provides the configuration instructions *within* the workflow steps themselves. Each workflow step provides detailed information about how to complete that step, which can include specifying variable values that are tailored for your system, performing a manual action, or executing an automated action. To gain familiarity with the overall process, you can read through the workflow steps just as you would read through a separate set of written instructions before you actually perform them.

The configuration workflow performs the setup for the following z/OS AI Framework components:

- The data collection engine
- The AI model server
- The AI Control Interface for IBM z/OS
- The AI Base Component for IBM z/OS

Note: In addition to configuring these z/OS AI Framework components with this workflow, you must also configure at least one AI-enhanced use case. Each use case provides its own configuration workflow similar to this one, as further described under each use case in [Part 3, “AI providers and use cases,” on page 77.](#)

The AI Framework for z/OS Configuration Workflow is shipped as an XML file known as a *workflow definition file*. Before you can use a workflow for the first time, you must create an instance of the workflow in z/OSMF from the workflow definition file. After you create the workflow instance, you can run the workflow to configure the z/OS AI Framework.

Plan ahead for multi-system support: Note the following points about configuring the z/OS AI Framework on multiple systems.

- In several places during the configuration of the z/OS AI Framework, you will be directed to export variables to a z/OS UNIX file. These export steps allow you to save your variable inputs as a property file so that those values can be automatically populated when configuring the z/OS AI Framework on another system. This will save you significant time and effort when you configure the z/OS AI Framework on additional systems.
- After completing the initial z/OS AI Framework configuration on one system, if you wish to configure the framework on additional systems, you must use a *separate* workflow to configure the z/OS AI Framework on each additional system, as described in [“Adding another z/OS system to your z/OS AI Framework configuration” on page 20.](#) You can do this right away, or you can wait to configure additional systems until you have had some time to test and become familiar with the z/OS AI Framework on the first system.
- The z/OSMF data directory must be mounted on a shared, global path, `/global/zosmf`, as described in [Step 2: Run job IZUMKFS to create the z/OSMF user file system in IBM z/OS Management Facility Configuration Guide.](#) This allows z/OS AI Framework components that are running on remote systems to access z/OSMF persistent data.
- IBM suggests that you share the started-task procedures (PROCs), as this will simplify the overall configuration process. If you do so, ensure that the specified ports match across all the systems in workflow step 2, “Collect input for all components to configure the AI Framework for IBM z/OS.”

Procedure

Complete these steps to create an instance of the AI Framework for z/OS Configuration Workflow from the workflow definition file and start the workflow to perform the actual configuration of the z/OS AI Framework. The actual configuration steps and actions are described in detail within the workflow itself.

1. Go to the z/OSMF Desktop for your system.
2. Double-click **Workflows**.
3. To create the workflow instance for the first time, click **Actions > Create Workflow**, then enter the following values:
 - a) In the **Location** field, select the system where the workflow definition file and variable input files are installed.
 - b) In the **Workflow definition file** field, enter the following path for the workflow definition file:

```
/usr/lpp/bcp/zosmf/workflows/AI_Framework_Config.xml
```

- c) Leave the **Workflow variable input file** field empty.
 - d) Click **Next**.
 - e) In the **Workflow name** field, enter the name that you want to use for the workflow.

This is the name that will appear in the list of workflows on the main Workflows page. The name must be unique among all the workflows in the **Workflow Name** column on the main Workflows page.

This documentation refers to this workflow as "Configure AI Framework for IBM z/OS," which is the default workflow name, but you can use any name you want as long as it is unique on your system.
 - f) Click **Finish**.
- The newly created workflow instance opens.
- Now that you have created an instance of the workflow in z/OSMF, you do not need to do it again on this system.
 - While using the workflow, if you want to pause and restart later, skip step 3 in its entirety and click the name of the workflow instance in the **Workflow Name** column to open it.
4. To begin the workflow, click the title of the first step, called "Introduction to AI Framework for IBM z/OS configuration workflow."

The step opens and displays the **General** tab.

5. Click the **Perform** tab to see detailed information about the step.

For this step, the information provides an overall description of what the configuration workflow does and an overview of the sequence of steps that you will be performing.

Important: Do not skip this step! The "Introduction" step contains important information and instructions that you need to know before proceeding with the workflow.

When you have finished reading the information, click **Finish** to complete the step.

Note: If you are new to workflows, before you proceed any further, you might find it helpful to review the information in [About workflows \(www.ibm.com/docs/en/zos/3.1.0?topic=overview-about-workflows\)](http://www.ibm.com/docs/en/zos/3.1.0?topic=overview-about-workflows) in IBM Documentation to learn more about how to navigate and work with workflows in z/OSMF.

6. Click each workflow step, in sequence, to work your way through the configuration process.

The instructions for each step are contained within the workflow. Click the **Perform** tab for each step to see the details. For an input field, you can also click the information icon next to the field to display more information about that field.

You will notice that steps often have substeps. Progress through the substeps in a similar manner until each step is complete.

Note: As described earlier, several workflow steps allow you to export variable inputs into a property file. If you plan to configure the z/OS AI Framework on additional systems, which requires running

a separate workflow, you can specify this file when you create the instance of that workflow to automatically populate the variable inputs, which will make the process to configure each additional system much faster and easier. For details, see [“Importing and exporting configuration variables in a property file”](#) on page 22.

At any time, you can click **Save** to save your progress. You can save and pause the workflow and continue later, and you can assign steps to other people (such as a security administrator, storage administrator, or network administrator), and track progress to complete the configuration.

Results

Upon completion of the workflow, you will have all the z/OS AI Framework components configured and available on a single z/OS system.

What to do next

- When you have successfully completed the z/OS AI Framework configuration workflow, you can configure an AI-enhanced use case to begin using the functionality that it provides. Each use case provides its own configuration workflow, which adds that use case to your overall z/OS AI Framework configuration on this system. These workflows are described for each use case in [Part 3, “AI providers and use cases,”](#) on page 77.
- After you have successfully completed the z/OS AI Framework configuration workflow and an AI-enhanced use case workflow, you can use the AI Control Interface for IBM z/OS to manage the AI-enhanced use case, as described in [Chapter 5, “Using the AI Control Interface for IBM z/OS,”](#) on page 27 and [Part 3, “AI providers and use cases,”](#) on page 77.
- If you want to configure the z/OS AI Framework on one or more additional z/OS systems, do not rerun this workflow. Instead, when you are ready to proceed, use the Configure AI Framework for z/OS for Additional Systems Workflow, as described in [“Adding another z/OS system to your z/OS AI Framework configuration”](#) on page 20.

Adding another z/OS system to your z/OS AI Framework configuration

Complete this task to add another z/OS system to your existing z/OS AI Framework configuration.

Before you begin

Be sure you have completed the initial z/OS AI Framework configuration, as described in [“Using the AI Framework for IBM z/OS Configuration Workflow”](#) on page 17.

About this task

The Configure AI Framework for IBM z/OS for Additional Systems Workflow guides you, step-by-step, through the process to add another z/OS system to your existing z/OS AI Framework configuration. This workflow is similar to the AI Framework for IBM z/OS Configuration Workflow that you used to perform the initial configuration, but has fewer steps because not every z/OS AI Framework component needs to be configured on each additional system.

Each workflow step provides detailed information about how to complete that step, which can include specifying variable values that are tailored for your system, performing a manual action, or executing an automated action.

This configuration workflow performs the setup for the following z/OS AI Framework components that are required on each additional system:

- The data collection engine
- The AI Base Component for IBM z/OS

Note: In addition to configuring these z/OS AI Framework components with this workflow, you must also configure at least one AI-enhanced use case on each additional system. Each use case provides its own configuration workflow similar to this one, as further described under each use case in [Part 3, “AI providers and use cases,”](#) on page 77

The Configure AI Framework for IBM z/OS for Additional Systems Workflow is shipped as an XML file known as a *workflow definition file*. Before you can use a workflow for the first time, you must create an instance of the workflow in z/OSMF from the workflow definition file. After you create the workflow instance, you can run the workflow to add a z/OS system to your z/OS AI Framework configuration.

Procedure

Complete these steps to create an instance of the Configure AI Framework for IBM z/OS for Additional Systems Workflow from the workflow definition file, and then run the workflow.

1. Go to the z/OSMF Desktop for your system.
2. Double-click **Workflows**.
3. To create the workflow instance for the first time, click **Actions > Create Workflow**, then enter the following values:
 - a) In the **Location** field, select the system where the workflow definition file and variable input files are installed.
 - b) In the **Workflow definition file** field, enter the following path for the workflow definition file:

```
/usr/lpp/bcp/zosmf/workflows/AddSystemForAI_Config.xml
```

- c) In the **Workflow variable input file** field, enter the path to the variable property file that you created during the initial configuration workflow.

Specifying the workflow variable input file enables the saved variable values from the initial configuration workflow to automatically populate the input variables for this workflow, making the process of completing this workflow much faster and easier. For details, see [“Importing and exporting configuration variables in a property file”](#) on page 22.
 - d) Click **Next**.
 - e) In the **Workflow name** field, enter the name that you want to use for this workflow.

This is the name that will appear in the list of workflows on the main Workflows page. The name must be unique among all the workflows in the **Workflow Name** column on the main Workflows page.

This documentation refers to this workflow as "Configure AI Framework for IBM z/OS for Additional Systems" but you can use any name you want as long as it is unique on your system.
 - f) Click **Finish**.

The newly created workflow instance opens.

 - Now that you have created an instance of the workflow in z/OSMF, you do not need to do it again on this system.
 - While using the workflow, if you want to pause and restart later, skip step 3 in its entirety and click the name of the workflow instance in the **Workflow Name** column to open it.
4. To begin the workflow, click the title of the first step in the workflow, called "Introduction to the AI Framework for IBM z/OS for additional systems workflow."

The step opens and displays the **General** tab.
 5. Click the **Perform** tab to see detailed information about the step.

For this step, the information provides an overall description of what the configuration workflow does and an overview of the sequence of steps that you will be performing.

Important: Do not skip this step! The "Introduction" step contains important information and instructions that you need to know before proceeding with the workflow.

When you have finished reading the information, click **Finish** to complete the step.

Note: If you are new to workflows, before you proceed any further, you might find it helpful to review the information in [About workflows \(www.ibm.com/docs/en/zos/3.1.0?topic=overview-about-workflows\)](http://www.ibm.com/docs/en/zos/3.1.0?topic=overview-about-workflows) in IBM Documentation to learn more about how to navigate and work with workflows in z/OSMF.

6. Click each workflow step, in sequence, to work your way through the configuration process.

The instructions for each step are contained within the workflow. Click the **Perform** tab for each step to see the details. For an input field, you can also click the information icon next to the field to display more information about that field.

You will notice that steps often have substeps. Progress through the substeps in a similar manner until each step is complete.

At any time, you can click **Save** to save your progress. You can save and pause the workflow and continue later, and you can assign steps to other people (such as a security administrator, storage administrator, or network administrator), and track progress to complete the configuration.

Results

Upon completion of the workflow, you will have the subset of z/OS AI Framework components that are required on the added system configured and available.

What to do next

You can use the AI Control Interface for IBM z/OS to manage the AI-enhanced use case for this system, as described in Chapter 5, “Using the AI Control Interface for IBM z/OS,” on page 27 and Part 3, “AI providers and use cases,” on page 77.

- When you have successfully completed the z/OS AI Framework configuration workflow, you can configure an AI-enhanced use case to begin using the functionality that it provides. Each use case provides its own configuration workflow, which adds that use case to your overall z/OS AI Framework configuration on this system. These workflows are described for each use case in Part 3, “AI providers and use cases,” on page 77.
- After you have successfully completed the z/OS AI Framework configuration workflow and an AI-enhanced use case workflow, you can use the AI Control Interface for IBM z/OS to manage the AI-enhanced use case, as described in Chapter 5, “Using the AI Control Interface for IBM z/OS,” on page 27 and Part 3, “AI providers and use cases,” on page 77.

Importing and exporting configuration variables in a property file

When configuring the AI Framework for IBM z/OS, you are prompted to supply a number of variable input values while using the initial AI Framework for IBM z/OS Configuration Workflow, AI provider use case workflows, and the workflows that they invoke. Many of these input values can be reused when configuring AI use cases and additional systems for the z/OS AI Framework. Thus, to avoid redundant prompting and simplify completion of the configuration process on additional systems, the workflows provide the ability to save, or export, those variables to a z/OS UNIX file called a *property file*.

During the initial step, “Collect input to configure the AI Framework,” you must specify a property file path in the **Configuration variables export absolute file path** field. At several points during the configuration workflow, there are data collection steps in which you must provide data for a variety of variables. Following the data collection steps, you will be able to export the variables to the property file at the previously specified path. Each export for a given system merges the variables into the same property file that will be used for all data collected by the configuration workflows on that specific system.

When you want to expand the z/OS AI Framework configuration to additional systems by using the Configure AI Framework for IBM z/OS for Additional Systems Workflow, you can use the exported property file to prime the variables by specifying in the **Workflow variable input file** field on the **Create Workflow** dialog the path to the property file that was created during the initial configuration workflow. Repeat this action for each workflow that is called from the “additional systems” workflow. Then, as you

make system-specific changes to the variable data on each additional system, you will again export the variables to a property file that is unique for that system.

The following key points summarize how to effectively use property files with the configuration workflows:

- For any given system, use the *same* property file to export variables for *all* configuration workflows for that system. All exported variables will be merged into the same file for that system.
- When adding another system to your z/OS AI Framework configuration:
 - To prime the variable values, import the values from the property file that you exported from the system that you initially configured. Specify the path for that property file in the **Workflow variable input file** field on the **Create Workflow** dialog for the "Configure AI Framework for IBM z/OS for Additional Systems" workflow on the new system.
 - To export variables from your configuration on the new system, specify a *different* path for the property file for that system in the **Configuration variables export absolute file path** field. Use this same path for all exports from all configuration workflows on the new system.
- As a general rule, you *import from* a property file that was exported on a previously configured system, and you *export to* a system-specific property file for the current system that you are configuring.

Avoiding AT-TLS configuration conflicts between the z/OS AI Framework configuration workflow and Network Configuration Assistant

The z/OS AI Framework configuration workflows generate AT-TLS policy statements. If you also use the z/OSMF Network Configuration Assistant (NCA), be aware that NCA overwrites the AT-TLS configuration, which can remove statements that were added by the z/OS AI Framework configuration workflows. There are 2 options to avoid losing the z/OS AI Framework AT-TLS policy statements:

Option 1

Instead of using the z/OS AI Framework workflow-generated statements as-is, use NCA to add those statements to the AT-TLS policy so that they will not be overwritten by NCA going forward.

Option 2

Add the z/OS AI Framework workflow-generated statements to an AT-TLS configuration file that NCA does not manage.

For option 2, check the contents of your PAGENT configuration file (as specified by **PAGENT_CONFIG_FILE**), which specifies 1 or 2 AT-TLS configuration files that contain AT-TLS rules (see [Policy Agent configuration files overview in z/OS Communications Server: IP Configuration Reference](#)):

- The **CommonTLSConfig** statement specifies a common AT-TLS configuration file that applies to all IP stacks in the LPAR.
- The **TTLSTLSConfig** statement specifies a stack-specific configuration file.

NCA manages 1 of these files. By adding the z/OS AI Framework rules to the file that is *not* managed by NCA, you can avoid having NCA overwrite them. (If a same-named rule exists in both files, the rule in the **TTLSTLSConfig** file is used.)

Example: Assume that your PAGENT configuration file contains the following statements:

```
TTLSTLSConfig /etc/cfgasst/tlsPol  
IPSecConfig /etc/cfgasst/ipsecpol
```

You can then create a new policy with the z/OS AI Framework AT-TLS statements in a new file, such as `/etc/cfgasst/zAIPol`, and update the policy to contain the following statements:

```
TTLSTLSConfig /etc/cfgasst/tlsPol  
CommonTTLSTLSConfig /etc/cfgasst/zAIPol  
IPSecConfig /etc/cfgasst/ipsecpol
```

If both **TTLSTLSConfig** and **CommonTTLSTLSConfig** are already specified, check to see which one is managed by NCA. To do that, from the NCA AT-TLS main panel, go to **Actions > Install configuration files**. The panel that appears indicates the most recent place where each file is installed.

Upgrading the z/OS AI Framework configuration from AI System Services for IBM z/OS 1.1 to 1.2

Complete this task to upgrade the z/OS AI Framework configuration from AI System Services for IBM z/OS 1.1 to 1.2.

AI System Services for z/OS 1.2 (also referred to as "AISS" in this documentation) contains IBM Z Common Data Provider 5.1 and Machine Learning for IBM z/OS 3.2 Core Edition. Do not use any mechanism other than this workflow to upgrade these products for use with the z/OS AI Framework.

Before you begin

This task is only applicable if you want to upgrade to AISS 1.2 and you have already completed the initial z/OS AI Framework configuration using AISS 1.1.

About this task

The AI Framework Upgrade AISS 1.2 Workflow guides you, step-by-step, through the process of upgrading from AI System Services for z/OS 1.1 to AI System Services for z/OS 1.2. This workflow is similar to the other workflows you have used to configure the z/OS AI Framework, but only includes steps required to upgrade the configuration to use AI System Services for z/OS 1.2.

Each workflow step provides detailed information about how to complete that step, which can include specifying variable values that are tailored for your system, performing a manual action, or executing an automated action.

The AI Framework Upgrade AISS 1.2 Workflow is shipped as an XML file known as a *workflow definition file*. Before you can use a workflow for the first time, you must create an instance of the workflow in z/OSMF from the workflow definition file. After you create the workflow instance, you can run the workflow to upgrade AI System Services for z/OS from 1.1 to 1.2 in your installation.

Procedure

Complete these steps to create an instance of the AI Framework Upgrade AISS 1.2 Workflow from the workflow definition file and start the workflow to perform the actual upgrade to AI System Services for z/OS 1.2 in your z/OS AI Framework configuration. The actual configuration steps and actions are described in detail within the workflow itself.

1. Go to the z/OSMF Desktop for your system.
2. Double-click **Workflows**.
3. To create the workflow instance for the first time, click **Actions > Create Workflow**, then enter the following values:
 - a) In the **Location** field, select the system where the workflow definition file is installed.
 - b) In the **Workflow definition file** field, enter the following path for the workflow definition file:

```
/usr/lpp/bcp/zosmf/workflows/AI_Framework_Upgrade_AISS_1_2.xml
```
 - c) In the **Workflow variable input file** field, enter the path to the variable property file that you created during the initial configuration workflow, if one is available.

Specifying the workflow variable input file enables the saved variable values from the initial configuration workflow to automatically populate the input variables for this workflow, making the process of completing this workflow much faster and easier. For details, see [“Importing and exporting configuration variables in a property file”](#) on page 22.
 - d) Click **Next**.
 - e) In the **Workflow name** field, enter the name that you want to use for the workflow.

This is the name that will appear in the list of workflows on the main **Workflows** page. The name must be unique among all the workflows in the **Workflow Name** column on the main **Workflows** page.

This documentation refers to the workflow as "AI Framework Upgrade AISS 1.2 Workflow," which is the default workflow name, but you can use any name you want as long as it is unique on your system.

f) Click **Finish**.

The newly created workflow instance opens.

- Now that you have created an instance of the workflow in z/OSMF, you do not need to do it again on this system.
- While using the workflow, if you want to pause and restart later, skip step 3 in its entirety and click the name of the workflow instance in the **Workflow Name** column to open it.

4. To begin the workflow, click the title of the first step, called "Introduction to AI Framework Upgrade AISS 1.2 Workflow."

The step opens and displays the **General** tab.

5. Click the **Perform** tab to see detailed information about the step.

For this step, the information provides an overall description of what the workflow does and an overview of the sequence of steps that you will be performing.

Important: Do not skip this step! The "Introduction" step contains important information and instructions that you need to know before proceeding with the workflow.

When you have finished reading the information, click **Finish** to complete the step.

Note: If you are new to workflows, before you proceed any further, you might find it helpful to review the information in [About workflows \(www.ibm.com/docs/en/zos/3.1.0?topic=overview-about-workflows\)](http://www.ibm.com/docs/en/zos/3.1.0?topic=overview-about-workflows) in IBM Documentation to learn more about how to navigate and work with workflows in z/OSMF.

6. Click each workflow step, in sequence, to work your way through the configuration process.

The instructions for each step are contained within the workflow. Click the **Perform** tab for each step to see the details. For an input field, you can also click the information icon next to the field to display more information about that field.

You will notice that steps often have substeps. Progress through the substeps in a similar manner until each step is complete.

At any time, you can click **Save** to save your progress. You can save and pause the workflow and continue later, and you can assign steps to other people and track progress to complete the workflow.

Results

Upon completion of the workflow, you will have upgraded your z/OS AI Framework configuration to AI System Services for z/OS 1.2.

Chapter 5. Using the AI Control Interface for IBM z/OS

After you have successfully configured the AI Framework for IBM z/OS and the AI provider use case, you can use the AI Control Interface for IBM z/OS to manage the use case. You can access AI Control Interface for z/OS from the **AI Control Interface** task on the z/OSMF desktop.

The following topics describe how to use the AI Control Interface for z/OS. This is the same information that is available as online help within the AI Control Interface for z/OS user interface; therefore, it is likely to be most helpful to read this information while referring to that interface.

Note: This information is also available as online help within z/OSMF and in *AI Control Interface* AI Control Interface (www.ibm.com/docs/zos/latest?topic=interface-ai-control) in IBM Documentation.

AI Control Interface

AI Control Interface for IBM z/OS automates and enhances z/OS system management tasks with prebuilt AI models.

With AI Control Interface, you can perform management actions to train models, update model settings, view the status of AI enablement, and check the health of the z/OS AI Framework.

Refer to the following links to learn how to use AI Control Interface:

AI-enabled use cases

The use cases that are available to configure for AI Control Interface. For more information, see [“AI use cases”](#) on page 28.

Get started

Learn how to get started for the first time using AI Control Interface. For more information, see [“Getting started in AI Control Interface”](#) on page 27.

Manage intelligent operations

Select an AI-enabled tool and begin working with the available use cases for your configured z/OS systems. For more information, see [“Managing intelligent operations”](#) on page 29.

Manage components in a system

View the available components that AI can manage for a selected AI-enabled tool in your z/OS systems, begin training, and edit AI mode for available components. For more information, see [“Managing components in a system”](#) on page 31.

Getting started in AI Control Interface


Learn how to get started when you use AI Control Interface for IBM z/OS for the first time.


To begin using AI Control Interface, select an AI-enabled tool from the list of available use cases on the **Get started** page.

If there are no use cases configured yet, a tile displays that there are no AI-enabled tools available. The tools must be enabled by a z/OS system programmer outside of AI Control Interface using one or more z/OSMF configuration workflows. After your z/OS system programmer has configured the use case that you want to use, you can click the **Refresh** button on the tile to check the current tool configuration.

When a function is enabled for your system, the name of the AI-enabled tool and a brief description of what use cases you can control with it is displayed. You can click anywhere on the tile to proceed to the **Manage** tab to activate AI support on a system.



Connection status

You can view the connection status of AI Control Interface by clicking the  icon in the main navigation bar.

If either the AI Base Component for IBM z/OS or the MLz Core is not connected, the icon displays as crossed as an indicator (). Clicking the icon displays a detailed connection error.

Notifications

A notification is a notice of some occurrence in the system that requires your awareness or response. A notification can be informational in nature, or it can be a request for action.

To access your notifications, click the Notifications icon () in the main navigation bar. When you have unread notifications, the icon is shown with a circle for emphasis (.

For some notifications, a hyperlink is provided to a message that gives additional details and actions.

Terms you should know

Understand terminology in AI Control Interface.

administrator

A z/OS system programmer who is an administrator for z/OS AI Control Interface. Administrators have access to the IZUADMIN security group.

AI-enabled tool

System functions configured for use with AI Control Interface. These providers contain prebuilt AI models to automate and enhance z/OS system management.

component

Individual z/OSMF objects in a system that can be configured for management by the AI model. The type of z/OSMF object is dependent on the associated use case for that system.

use case

An AI-enhanced task that can be performed on a selected system.

AI use cases

AI use cases are system functions that can be configured for use with AI Control Interface.

The following use cases are available for you to get started with today.

Note: Additional tools are planned for future releases, check back for updates.

z/OS Workload Management (WLM)

Workload Management (WLM) monitors a z/OS system and dynamically provides resources to incoming items of work based on predefined goals.

Use case: AI-powered WLM batch initiator management

Using sophisticated workload trends, you can proactively initiate resource assignment so that a workload can start right away.

Component: Batch service classes

A service class is a group of work with similar performance goals, resource requirements, and business importance. z/OS Workload Management manages each group of work according to the performance goal assigned to the service class, and the business importance assigned to that performance goal.

The batch service classes that are displayed in AI Control Interface are a subset of the total service classes on a given system. These available batch service classes have usable SMF data that qualifies them to be enhanced by AI-powered WLM batch initiator management.

Training a system

Basic concepts to consider when you train systems in AI Control Interface for IBM z/OS.

Model training is a long-running operation, so there are some factors to consider before you decide to initiate the training process.

Initiating training

To start training a selected system, at least one component must be configured and discovered by AI Control Interface. Model training runs on a system level, and includes all components that are discovered in that system and are listed in the components table. For more information, see "Managing components in a system."

You cannot train multiple systems concurrently. One system must complete the training process before you are able to start another in either a different system or a different AI-enabled tool. The **Train System** button is disabled across all table instances while a training process is running.


The **Train System** button is also disabled if any rows are selected in the components table. Deselect all components by individual row or by using the **Cancel** batch action in the table to proceed.

During training

You do not need to keep AI Control Interface open while training is running. Training proceeds in the background while your z/OSMF server is running.

You are able to switch the AI-mode of a successfully trained component while another training process is running. Only components that were previously successfully trained can be put into *Enable* mode. For more information, see "Managing components in a system."

When training completes

When a training process completes, messages are briefly displayed as notifications. These notifications are also accessible from the Notifications icon () in the main navigation bar, and persist until they are dismissed.

If a component fails during training, it is automatically switched to *Disabled* since the model is not able to make prediction results on a failed component.

Retraining

AI-powered WLM batch initiator management use case

To ensure that you are working with current and accurate system data, the recommended time period to retrain a system is thirty days. If a system is not retrained within that time, a **Needs attention** status tag is displayed in the AI mode column.

Managing intelligent operations

Use the **Manage** tab of AI Control Interface to select an AI-enabled tool and begin working with the available use cases that are configured for your z/OS systems.

The **Manage** tab is composed of the **AI-enabled tools** selection list and the current enablement table of the tool.

AI-enabled tools panel

The **AI-enabled tools** panel displays all available providers on which you can enable AI. The panel can be collapsed by using the chevron icon.

For more information about the tools that are available for AI-enablement in AI Control Interface, see "AI-enabled tools."

Systems table

The systems table name reflects the selected AI-enabled tool. The latest timestamp that was generated when the table of contents were refreshed is displayed.

The table is populated with the configured systems that have z/OS 3.1 installed and the selected AI-enabled tool enabled. The local system that has z/OSMF installed on it is always listed at the beginning of the table.

If no AI-enabled tools are configured yet, the table displays an empty state. You can click the **Refresh** button to check whether the tool configuration is complete. The refresh timestamp updates to reflect that the table contents were refreshed.

You can use the Search and Filter functions to locate a specific system within the table. You can filter and sort the table by AI status.

After training, any systems with a status of *Needs attention* are sorted to the beginning of the table, until a user applies an AI status of their choosing. This AI status selection persists for each user that is logged in. You can hover over each tag in the AI status column for a detailed explanation.

Click anywhere on a row to view more details about a component. The [Table 3 on page 30](#) provides more context about each column.

Table 3. Columns in the Systems table. These columns appear in the Systems table regardless of which option is selected from the **AI-enabled tools** panel.

Column	Description
System	The name of the system that can be configured for management by the AI model.
Use case	AI-enhanced tasks that are available to perform on the selected system.
AI status	<p>The current condition of the AI model used to perform the use case.</p> <p>Ready The system is newly available and can be trained with AI Control Interface.</p> <p>Needs attention It is recommended that you retrain the model for the system. The recommended time period for retraining is within thirty days.</p> <p>Inactive The system was trained, but none of the system's components are being managed by AI.</p> <p>Active The system has been trained and at least one of the system's components is being managed by AI.</p>
Components	<p>The current count and type of z/OSMF objects in the system that can be configured for management by the AI model. The type of z/OSMF object is dependent on the associated use case for that system.</p> <p>The value of this column is <i>Searching...</i> while AI Control Interface searches a selected system for enabled components when it is first added.</p> <p>For more information, see "AI-enabled tools."</p>

Managing components in a system

You can use this view to display the components on the selected system that can be managed by AI, begin training, and edit AI mode.

The components table reflects the components of a system that AI can manage for a selected AI-enabled tool in AI Control Interface. A description of the use case and components for the selected AI-enabled tool is provided, along with the timestamp that was generated when the table contents were refreshed. The latest timestamp that was generated when the training process was trained is also displayed if the system was previously trained.

If the system is searching for instances of the component, or no components are found after the search has completed, the table displays an empty state. You cannot initiate training until at least one of the component types is configured for AI Control Interface.

You can use the Search and Filter functions to locate a specific component within the table. You can filter and sort the table by training status and AI mode.

If another component is added after the system was trained, that component appears at the beginning of the table with a training status of *Not yet trained*, with Performance metrics *Not available* and AI mode *Disabled*.

After training completes, any components that have a training status of *Failed* are automatically sorted to the beginning of the table. You can hover over the status to see a detailed reason as to why training failed.

Columns in the components table

Some columns are persistent across all available use cases, while others are dynamically generated depending on the components of the system that AI can manage in the selected use case. [Table 4 on page 31](#) and [Table 5 on page 32](#) provide more context about each column.

Table 4. Persistent columns in the components table

Column	Description
Training status	A record of AI performance against actual values from simulating.
	Not yet trained The AI model for this component's system is not trained yet.
	Training An AI model is training to perform a use case for this component.
	Trained The component was successfully trained.
	Failed Training failed for this component. Hover over the status for a detailed message that explains why training failed. The AI mode for this component is automatically switched to <i>Disabled</i> , since the model is not able to make predictions for a failed component.
AI performance metrics	The current condition of the AI model used to perform the use case.
	Not available No AI performance metrics are available for this component.
	Check results AI performance metrics are available to display how the model would perform if it were enabled. Click the link to open a modal window with more details.
	Not applicable This component is enabled, so simulation-based AI performance metrics are not available.

Table 4. Persistent columns in the components table (continued)

Column	Description
AI mode	<p>The current state of AI operations on the component.</p> <p>Disabled The AI model is not operating a use case for this component. Simulation-based AI performance metrics are not available.</p> <p>Simulating The AI model is not operating a use case for this component, but it is generating predictions. Click <i>Check results</i> in the AI Performance metrics column to see how the model would have performed if it were enabled.</p> <p>Enabled The AI model is operating a use case for this component. Simulation-based performance metrics are not available.</p>

Table 5. Dependent columns in the components table. The content of this column is dynamically generated based on the option that was selected in the AI-enabled tools panel.

Column name	Associated AI-enabled tool	Description
Batch service classes	Workload Management	A group of work with similar performance goals, resource requirements, and business importance. Enhance batch service classes with an AI model for AI-powered WLM batch initiator management. For more information, see "AI-enabled tools."

Table 6. Targeted actions for the components table

Action	Description
Enable	Allows the AI model to operate the use case for the selected AI-enabled tool on the selected system. For more information, see "AI-enabled tools."
Simulate	Allows the AI model to make predictions about the use case for the selected AI-enabled tool on the selected system, but the model does not operate the use case. The selected AI-enabled tool continues to function without AI. You can check that the results after simulating is completed to see how the AI model would perform.
Disable	<p>Stops the AI model from operating the use case on the selected system. The selected AI-enabled tool continues to function without AI.</p> <p>Note: The AI-enabled tool continues to function as usual with no gaps in processing if you choose to disable the AI model.</p>

Table 7. Untargeted actions for the components table

Action	Description
Train system	Trains the model for the selected system. For more information, see "Training a system."

Review AI simulation

The **Review AI simulation** modal displays a visualization for a selected service class after training. The chart displays a comparison of the predicted number of initiators needed to the actual number of initiators that were started by WLM over the given time period.

The modal name reflects the selected service class.

The chart renders the performance data from the last 24 hours by default. A loading animation displays until the data is retrieved and rendered. If the data cannot be retrieved for any reason, the chart displays as an empty state with a corresponding error message.

You can refer to the information provided in the **How do I read this graph?** tooltip for additional context of what to look for in the chart behavior.

Chart interactions

There are multiple options available to select and display a customized range of data in the chart. [Table 8](#) on page 33 and [Table 9](#) on page 33 provide more context about each column.

Table 8. Controls in the AI simulation chart

Column	Description
Download plot as a png	Save the chart as a PNG image file with filename <code>system_component_mode.png</code> to the default path for the browser.
Zoom	Define an area to select by clicking and dragging the chart with your mouse. The point of your initial click defines the first boundary, and the point where the mouse is released defines the ending boundary. You can select a range before or after your initial mouse click. The chart visually defines the boundary with a darker background color for the excluded sections outside of the boundary lines. This interaction is selected by default.
Pan	Pan across the horizontal axis by clicking and dragging across the chart. Double-click to reset the pan back to the original view.
Zoom in	Zoom in by clicking the icon.
Zoom out	Zoom out by clicking the icon.
Zoom bar	Define a range by clicking and dragging the boundary lines on each side of the bar.
Legend	Toggle the data to display on the chart by clicking the label of a plot line in the legend. Double-click on a label to display only that selected data type on the chart.

Sidebar interactions

Table 9. Fields in the component sidebar

Action	Description
Service class	Switch to another available service class from the drop down list.

Table 9. Fields in the component sidebar (continued)

Action	Description
Start	<p>Select the date and time to begin the range of data to retrieve, within the last 40 days to the present time.</p> <p>Start date Select a start date for the data range by clicking on a date in the calendar menu, navigating to a date by using the arrow keys and then pressing Enter, or manually entering the date into the text input field . The calendar menu reflects the maximum time range that you can select, and any dates beyond the last 40 days cannot be selected. If you manually enter a date that is outside of the range, an error message appears on the field. If you click Apply with an invalid date in the field, updated data does not render in the chart.</p> <p>If you use the Last 24 hrs button to reset the view, this value is refreshed to the date 24 hours before current date.</p> <p>Start time By default, the timezone reflects the z/OS server's timezone. If you select Last 24 hrs, this value is refreshed to the time 24 hours before the current time.</p> <p>Clock Display the result in 24 hour or 12 hour time according to what is set in your browser.</p>
End	<p>Select the date and time to end the range of data to retrieve, within the last 40 days up to the present time.</p> <p>End date Select an end date for the data range by clicking on a date in the calendar menu, navigating to a date by using the arrow keys and then pressing Enter, or manually entering the date into the text input field. The calendar menu reflects the maximum time range that you can select, and any dates beyond the present day cannot be selected. If you manually enter a date that is outside of the range, an error message appears on the field. If you click Apply with an invalid date in the field, updated data does not render in the chart.</p> <p>End time By default, the timezone reflects the z/OS server's timezone. If you select Last 24 hrs, this value is refreshed to the time 24 hours before the current time.</p> <p>Clock Displays the result in 12 hour time. If the Start time is set to 24 hour, the clock is hidden.</p>

The actions are described in the following tables:

- [Targeted actions](#). Actions that apply to selections in the component sidebar fields.
- [General actions](#). Actions that apply to the entire simulation chart. No selection is required.

Table 10. Targeted actions

Action	Description
Apply	Submit changes made to the any fields made in the sidebar fields, and render the chart with the selected component and date range.

Table 11. General actions

Action	Description
View last 24hrs	Renders the chart to the default range of the last 24 hours.

Chapter 6. Operations

The following information describes how to operate various components of AI Framework for IBM z/OS.

Operating IBM Z Common Data Provider

The following IBM Z Common Data Provider (ZCDP) components are required by the AI Framework for IBM z/OS:

- System data engine

The system data engine is the data gatherer which collects SMF data in near real time. It can also process SMF historical data in batch. The default started task procedure name for the system data engine is HBOAISDE.

- Data streamer

The data streamer receives data from the system data engine and writes it to the EzNoSQL database. The default started task procedure name for the data streamer is HBOAIDS.

Starting the ZCDP components

Issue the following z/OS system commands in the order shown to start the ZCDP components:

```
START HBOAIDS[,DB=vsam_dsn_name]  
START HBOAISDE[,RES=smf_inmem_name]
```

The DB and RES parameters are optional and can be omitted if the values in the started task procedure JCL are correct for the LPAR. These parameters are intended for use when the started task procedure is shared among LPARs in a sysplex.

Stopping the ZCDP components

Issue the following z/OS system commands in the order shown to stop the ZCDP components:

```
STOP HBOAISDE  
STOP HBOAIDS
```

Operating AI Base Component for z/OS

Normally, the AI Base Component for IBM z/OS automatically starts during system startup, if you have successfully completed the steps in the AI Framework for IBM z/OS Configuration Workflow. However, if necessary, you can use the following z/OS system commands to manually start and stop the z/OS AI Base.

Starting AI Base Component for IBM z/OS (AIBPROC)

Use the START *aibproc* command to start AI Base Component for IBM z/OS.

AI Base Component for IBM z/OS does not start automatically and must be started before any application can use its functions.

Syntax

The following syntax is used to start AI Base Component for IBM z/OS:

```
S aibproc[,ASCBV31=YES,REUSASID=YES]
```

Parameters

The following parameters are used for starting AI Base Component for IBM z/OS:

aibproc

The name of the started procedure used to start AI Base Component for IBM z/OS. The default procedure has the name AIBPROC.

,ASCBV31=YES

Indicates that the *aibproc* address space is in 31-bit addressable storage.

For more information, see the description of the ASCBV31 parameter in [Starting a system task from a console in z/OS MVS System Commands](#).

,REUSASID=YES

Indicates that a reusable ASID is assigned to the *aibproc* address space.

For more information, see the description of the REUSASID parameter in [Starting a system task from a console in z/OS MVS System Commands](#).

Stopping AI Base Component for IBM z/OS (AIBPROC)

Use the STOP *aibproc* command to stop AI Base Component for IBM z/OS.

Syntax

The command syntax is:

```
P aibproc
```

Parameters

The parameter is:

aibproc

The name of the started procedure that was used to start AI Base Component for IBM z/OS. The default procedure ships with the name AIBPROC.

Chapter 7. Performance considerations

Consider the following suggested actions for dealing with performance-related situations with AI Framework for IBM z/OS.

Tip: You can find additional practical guidance about z/OS AI Framework performance in the white paper, [IBM z/OS 3.1 AI-Powered WLM Batch Initiator Management Resource and Tuning Guidelines](http://www.ibm.com/downloads/cas/A6LG1M38) (www.ibm.com/downloads/cas/A6LG1M38).

Avoiding zIIP-eligible work executing on general purpose CPs

To reduce the incidence of zIIP-eligible work executing on general purpose CPs, you can use the **Honor Priority** parameter for a WLM service class or the **IIPHONORPRIORITY** statement in the IEAOPTxx member of PARMLIB, or both.

- At the system level, setting IIPHONORPRIORITY=NO in IEAOPTxx means that general purpose CPs will not process zIIP-eligible work *unless it is necessary* to resolve resource contention with non-zIIP eligible work. This is fully described in [IEAOPTxx \(OPT parameters\)](#) in *z/OS MVS Initialization and Tuning Reference*.
- At the service class level, setting **Honor Priority** to NO for a WLM service class means that zIIP-eligible work in that service class does not overflow to general purpose CPs when there is insufficient zIIP capacity for the workload in the system, regardless of the setting for **IIPHONORPRIORITY** in IEAOPTxx. General purpose CPs may still help when it is necessary to resolve resource contention with general purpose CP work. This is fully described in [Honor priority](#) and [Reviewing parameter settings](#) in *z/OS MVS Planning: Workload Management*.

Chapter 8. Service and support considerations

The following topics discuss considerations related to software service and support for AI Framework for IBM z/OS.

Fix category (FIXCAT) for service updates

The fix category (FIXCAT) for PTFs related to AI Framework for IBM z/OS is `IBM.z/OS.AIFramework`.

Service updates might require you to train again

Even if your components have already completed training, after you apply service that contains an updated AI training application, you must initiate training via the AI Control Interface for IBM z/OS to train the AI model again based on the updated training application. PTFs that require you to train again will include a `++HOLD` for `ACTION` in the PTF cover letter.

Restarting the z/OS AI Base component

Be aware that diagnostic data for AI Base Component for IBM z/OS (AIB) may be lost when AIB ends. If you experience an issue that requires you to restart AIB, you might wish to take a dump and collect any job output prior to stopping and restarting AIB in order to preserve any data that might be needed for diagnosis.

Chapter 9. Reference information for system programmers

The following topics provide general reference information about AI Framework for IBM z/OS, such as messages, codes, and diagnosis information, that might be useful for system programmers.

You can find additional reference information that is specific to a particular AI use case under that use case in [Part 3, “AI providers and use cases,”](#) on page 77.

MVS system messages

AIB messages

The following messages are issued by AI Framework for IBM z/OS.

AIB0016I	SYSAIB CTRACE DEFINITION FAILED. RC=<i>rc</i>, RSN=<i>rsn</i>
-----------------	--

Explanation

The system cannot define the SYSAIB component trace.

In the message text:

rc

The return code provided by the CTRACE DEFINE macro.

rsn

The reason code provided by the CTRACE DEFINE macro.

System action

The system runs without the SYSAIB component trace.

Operator response

Notify the system programmer.

System programmer response

Search problem reporting data bases for a fix for the problem. If no fix exists, contact the IBM Support Center.

Problem determination

For information about the CTRACE return and reason codes, see [CTRACE macro](#) in *z/OS MVS Programming: Authorized Assembler Services Reference ALE-DYN*.

Source

AI Base Component for IBM z/OS

Module

AIBCTIT

Routing code

2, *

Descriptor code

5

AIB0017I	COMPONENT TRACE PARMLIB OPTION <i>optname</i> IS NOT VALID
-----------------	---

Explanation

The system encountered an incorrect option in the CTIAIBxx parmlib member that had been specified on a prior TRACE CT command.

In the message text:

optname

The specified option that is incorrect.

System action

The system does not start the requested component trace. Processing continues with the next option specified.

Operator response

Notify the system programmer.

System programmer response

Examine the options near the indicated character string for a misspelling or other error. Correct the error in the parmlib member before reissuing the TRACE CT command.

Source

AI Base Component for IBM z/OS

Module

AIBCTSSM

Routing code

*

Descriptor code

5

AIB0018I	COMPONENT TRACE PARMLIB MEMBER <i>parmlibmem</i> WAS NOT FOUND, COMPONENT DEFAULTS ARE USED
-----------------	--

Explanation

The system did not find the CTIAIBxx parmlib member.

In the message text:

parmlibmem

The parmlib member that was not found.

System action

The system continues with component trace defaults.

Operator response

Notify the system programmer.

System programmer response

Determine why the indicated CTRACE parmlib member was not found. Ensure that the parmlib member exists and contains the desired options.

Source

AI Base Component for IBM z/OS

Module

AIBCTIT

Routing code

2, *

Descriptor code

5

AIB0019I	AN ERROR OCCURRED WHILE PROCESSING ARM <i>reason</i> REQUEST, RC=<i>rc</i>, RSN=<i>rsn</i>
-----------------	---

Explanation

The AI Base Component for IBM z/OS encountered an error while processing an ARM request.

In the message text:

reason

One of the following values:

REGISTER

Request register with ARM failed.

READY

Request ready with ARM failed.

DEREGISTER

Request deregister with ARM failed.

rc

The return code provided by the IXCARM macro.

rsn

The reason code provided by the IXCARM macro.

System action

The AI Base Component for IBM z/OS initialization continues if a register request or a ready request fails. The AI Base Component for IBM z/OS shutdown continues if a deregister request fails. For errors during register and deregister, an ABEND and dump can accompany this message.

Operator response

Notify the system programmer.

System programmer response

Search problem reporting data bases for a fix for the problem. If no fix exists, contact the IBM Support Center.

Source

AI Base Component for IBM z/OS

Module

AIBINIT

Routing code

2, 10, *

Descriptor code

4

AIB0020E

FAILURE TO CONNECT WITH THE MODEL SERVER: *reason*

Explanation

The AI Base Component for IBM z/OS failed to establish a connection to the model server. This could have been the first attempt to connect or a loss of communication after previously having been connected.

In the message text:

reason

One of the following reasons:

AUTHORIZATION ISSUE

An authorization error was encountered.

COMMUNICATION FAILURE

Communication was not established or was disrupted.

System action

The system attempts to reconnect. If the model server is temporarily not operational, periodic attempts will be made to reestablish connectivity.

Operator response

None.

System programmer response

Ensure that the model server is operational. If the model server will not be operational, consider stopping the AI Base Component for IBM z/OS. If the model server is operational, it is possible that some authentication failure was encountered. Verify that security is properly set up and credentials have not expired.

Source

AI Base Component for IBM z/OS

Module

AIBINIT

Routing code

1, 2, 10

Descriptor code

3, 7

AIB0021I	AN ERROR OCCURRED WHILE PROCESSING A CONFIGURATION FILE. <i>reason</i>
-----------------	--

Explanation

An error occurred while accessing the file system.

In the message text:

reason

One of the following reasons:

UNABLE TO ACCESS THE SPECIFIED DIRECTORY PATH

While attempting to access the configuration file, an error occurred. Check the name of the path provided on the started procedure. Make sure it is mounted and accessible.

UNABLE TO ACCESS FRAMEWORK.JSON FILE

While attempting to access the configuration file, an error occurred. Make sure it is mounted and accessible. Ensure that the required `framework.json` file is available.

UNABLE TO ACCESS AIM.JSON FILE

While attempting to access the configuration file, an error occurred. Make sure it is mounted and accessible. Ensure that the required `aim.json` file is available.

System action

The AI Base Component for IBM z/OS terminates.

Operator response

Notify the system programmer.

System programmer response

Ensure that the path provided to the started procedure for AI Base Component for IBM z/OS is mounted and accessible and that the required configuration files are available; then, restart the procedure.

Source

AI Base Component for IBM z/OS

Module

AIBINIT, AIBTKMRB

Routing code

1, 2, 10

Descriptor code

4

AIB0022E

***procname* IS DELAYING FOR UNIX SYSTEM SERVICES AVAILABILITY**

Explanation

The AI Base Component for IBM z/OS must use z/OS UNIX System Services (z/OS UNIX) but it is not yet available on this system.

In the message text:

procname

The name of the procedure used to start the AI Base Component for IBM z/OS.

System action

The system will delay and check again until z/OS UNIX is available.

Operator response

None.

System programmer response

Ensure that z/OS UNIX is available before starting the AI Base Component for IBM z/OS. This might not be an issue because the AI Base Component for IBM z/OS automatically waits until z/OS UNIX is available. If you determine that this is an issue, consider using automation to start the AI Base Component for IBM z/OS after z/OS UNIX is up and available.

Source

AI Base Component for IBM z/OS

Module

AIBINIT

Routing code

1, 2, 10

Descriptor code

3, 7

AIB0023I

**UNKNOWN ERROR ENCOUNTERED. FUNCTION=*functioncode*,
RC=*resultcode*, RSN=*reasoncode*, STATUS=*statuscode*, DG=*diagcode***

Explanation

The system encountered an unexpected error.

In the message text:

functioncode

The function code.

resultcode

The result code.

reasoncode

The reason code.

statuscode

The status code.

diagcode

The diagnostic code.

System action

AI Base Component for IBM z/OS may terminate, depending on where the error was encountered.

Operator response

Notify the system programmer.

System programmer response

Contact IBM Support.

Source

AI Base Component for IBM z/OS

Module

AIBINIT, AIBTKMRB

Routing code

1, 2, 10

Descriptor code

4

AIB0024E

procname IS DELAYING FOR ENF SIGNALING: reason

Explanation

The AI Base Component for IBM z/OS is unable to signal to system components about its availability.

In the message text:

procname

The name of the procedure used to start AI Base Component for IBM z/OS .

reason

One of the following:

TOO MANY OUTSTANDING ENF SIGNALS

The ENF process table is full. No further requests can be accepted until ENF processes some of the requests already queued.

ENF HAS NOT BEEN INITIALIZED

ENF has not been initialized on the system.

INSUFFICIENT STORAGE AVAILABLE

The system cannot obtain sufficient storage for the request.

System action

The system will delay and try again.

Operator response

None.

System programmer response

If the condition persists, search for other system processes which may be affecting ENF processing. Search problem reporting databases for a fix for the problem. If no fix exists, contact the IBM Support Center.

Source

AI Base Component for IBM z/OS

Module

AIBINIT

Routing code

1, 2, 10

Descriptor code

3, 7

AIB0099I	DIAGNOSTIC INFORMATION <i>reason</i> <i>diagnostic-data</i> END OF DIAGNOSTIC INFORMATION
-----------------	---

Explanation

Diagnostic data is present and is written to the log.

In the message text:

reason

One of the following reasons:

(TYPE 1:)

Diagnostic data from message area.

(TYPE 2:)

Diagnostic data from inference response area.

diagnostic-data

One or more lines of diagnostic data.

System action

The system continues.

Operator response

None.

System programmer response

Determine if the diagnostic information can help you resolve the issue. If you cannot resolve the issue, contact IBM Support.

Source

AI Base Component for IBM z/OS

Module

AIBINIT, AIBTKMRB

Routing code

10

Descriptor code

4

AIB0101I	AI BASE COMPONENT FOR IBM Z/OS IS ALREADY ACTIVE
-----------------	---

Explanation

A request to start AI Base Component for IBM z/OS was received; however, it is already active.

System action

The system ignores the start request.

Operator response

None.

System programmer response

None.

Source

AI Base Component for IBM z/OS

Module

AIBINIT

Routing code

2, *

Descriptor code

5

AIB0102I	AI BASE COMPONENT FOR IBM Z/OS MUST BE STARTED AS A STARTED TASK. JOB <i>jobname</i> ENDS
-----------------	--

Explanation

The named batch job attempted to start AI Base Component for IBM z/OS. It must be started as a STARTED task.

In the message text:

jobname

The name of the batch job.

System action

The system ends the job.

Operator response

Enter a START operator command to start AI Base Component for IBM z/OS.

System programmer response

None.

Source

AI Base Component for IBM z/OS

Module

AIBINIT

Routing code

1, 2, 10

Descriptor code

None.

AIB0103I

***procname* INITIALIZATION COMPLETE**

Explanation

Initialization is now complete for AI Base Component for IBM z/OS.

In the message text:

procname

The name of the procedure that was used to start AI Base Component for IBM z/OS.

System action

System processing continues.

Operator response

None.

System programmer response

None.

Source

AI Base Component for IBM z/OS

Module

AIBINIT

Routing code

2, *

Descriptor code

5

AIB0104I *procname* SHUTDOWN IN PROGRESS

Explanation

AI Base Component for IBM z/OS processing is ending in response to a system command or as a result of a system problem.

In the message text:

procname

The name of the procedure that was used to start AI Base Component for IBM z/OS.

System action

AI Base Component for IBM z/OS starts shutting down.

Operator response

Contact your system programmer if there are error messages accompanying this message.

System programmer response

No action is required if this is a normal ending of AI Base Component for IBM z/OS. If this is an error situation, see the messages associated with the error.

Source

AI Base Component for IBM z/OS

Module

AIBINIT

Routing code

2

Descriptor code

None.

AIB0105I *procname* SHUTDOWN IS COMPLETE

Explanation

AI Base Component for IBM z/OS processing has ended in response to a system command or as a result of a system problem.

In the message text:

procname

The name of the procedure that was used to start AI Base Component for IBM z/OS.

System action

AI Base Component for IBM z/OS ends.

Operator response

Contact your system programmer if there are error messages accompanying this message.

System programmer response

No action is required if this is a normal ending of AI Base Component for IBM z/OS. If this is an error situation, see the messages associated with the error.

Source

AI Base Component for IBM z/OS

Module

AIBINIT

Routing code

2

Descriptor code

None.

AIB0106I	A STOP COMMAND FOR AIB INSTANCE <i>procname</i> HAS BEEN ACCEPTED.
-----------------	---

Explanation

AI Base Component for IBM z/OS accepted the STOP command and began stop processing.

In the message text:

procname

The name of the procedure that was used to start AI Base Component for IBM z/OS.

System action

AI Base Component for IBM z/OS begins stop processing.

Operator response

None.

System programmer response

None.

Source

AI Base Component for IBM z/OS

Module

AIBTKCMD

Routing code

2, *

Descriptor code

5

AIB0109E	AI BASE COMPONENT FOR IBM Z/OS ADDRESS SPACE USER IDENTITY DOES NOT HAVE ACCESS TO UNIX SYSTEM SERVICES
-----------------	--

Explanation

The user security profile is either missing or is incomplete, or the OMVS segment is not defined for this user.

System action

AI Base Component for IBM z/OS ends.

Operator response

None.

System programmer response

Fix the security profile or define the OMVS segment, then restart AI Base Component for IBM z/OS.

Source

AI Base Component for IBM z/OS

Module

AIBINIT

Routing code

1, 2, 10

Descriptor code

7, 11

AIB0110I	<i>procname</i> REQUIRES A PARM= OR PARMDD= THAT SPECIFIES A VALID PATH FOR CONFIGURATION FILES.
-----------------	---

Explanation

The proc to start AI Base Component for IBM z/OS requires a parameter that specifies the path where the configuration files can be found. The path must be specified and must be less than 1023 bytes long. Given that file names must also be appended to the specified path, it must be smaller than the maximum z/OS UNIX System Services path size limit (for example, /framework.json must be added, which would reduce the path to a limit of 1008 bytes).

In the message text:

procname

The name of the procedure used to start AI Base Component for IBM z/OS.

System action

AI Base Component for IBM z/OS ends.

Operator response

None.

System programmer response

Change the procedure to include a PARM or PARMDD parameter that specifies the path, and restart the procedure.

Source

AI Base Component for IBM z/OS

Module

AIBINIT

Routing code

2, 10

Descriptor code

4

AIB0117I *procname* **INITIALIZATION STARTING**

Explanation

Initialization is starting for AI Base Component for IBM z/OS.

In the message text:

procname

The name of the procedure that was used to start AI Base Component for IBM z/OS.

System action

System processing continues.

Operator response

None.

System programmer response

None.

Source

AI Base Component for IBM z/OS

Module

AIBINIT

Routing code

2, *

Descriptor code

5

AIB0120I	AI BASE COMPONENT FOR IBM Z/OS IS ENDING DUE TO EXCESSIVE ERRORS
-----------------	---

Explanation

AI Base Component for IBM z/OS ends.

System action

The system runs without AI Base Component for IBM z/OS.

Operator response

Notify the system programmer.

System programmer response

Search problem reporting databases for a fix for the problem. If no fix exists, contact the IBM Support Center.

Source

AI Base Component for IBM z/OS

Module

AIBINIT

Routing code

1, 2, 10

Descriptor code

None.

AIB1215E	AN ERROR OCCURRED DURING PROCESSING <i>provider</i> <i>usecase</i> <i>model</i>
-----------------	---

reason

Explanation

An error was encountered while attempting to perform processing related to the specified model.

In the message text:

provider

The provider of the model that encountered the error.

usecase

The use case of the model that encountered the error.

model

The name of the model that encountered the error.

reason

One of the following reasons:

COMMUNICATION WITH MODEL SERVER IS DISRUPTED

During processing, a communication error was detected. The model provider was notified.

AUTHENTICATION ERROR ENCOUNTERED

An authentication error was encountered. The request was not successful. The model provider was notified, and processing will be suspended until the authentication issue is resolved.

THE MODEL IS NOT DEPLOYED

The model server indicated that the model was not found.

UNEXPECTED ERROR ENCOUNTERED

An unexpected error was encountered. The request may not have been successful. A companion message, AIB0023I, may have been issued which contains additional diagnostic information.

System action

The processing was unsuccessful. The next action may be up to the provider of the model, or if a configuration error was encountered, the system programmer might need to resolve the issue in the AI Framework for IBM z/OS Configuration Workflow in z/OSMF.

Once the issue is resolved, processing can resume.

Operator response

Notify the system programmer.

System programmer response

The error might be transient or permanent. The owning application (provider) that requested the processing was notified of the error and, if possible, might take corrective action. If the problem is with the configuration, use the AI Framework for IBM z/OS Configuration Workflow to resolve the issue.

Problem determination

Refer to the component documentation.

Source

AI Base Component for IBM z/OS

Module

AIBTKMRB

Routing code

1, 2, 10

Descriptor code

3, 7

AIBH0002I

hzsckname check was entered with an unknown function code of
fncode

Explanation

The check was not coded to recognize the specified function code. The function code was ignored.

System action

System processing continues.

Operator response

None.

System programmer response

None.

Module

AIBHCCHK, AIBHMSG

AIBH0100I

hzsckname check was entered with an unknown entry code of
entrycode

Explanation

An unexpected error occurred.

System action

The check is not processed.

Operator response

Notify the system programmer.

System programmer response

Refresh the check. If the problem persists, contact IBM Support with the information provided in this error message.

Module

AIBHCCHK, AIBHMSG

AIBH0300I

The installed MLz Core version is at a recommended level.

Explanation

Check successful. Machine Learning for IBM z/OS Core Edition (MLz Core) is at the recommended level.

System action

System processing continues.

Operator response

None.

System programmer response

None.

Module

AIBHCCHK, AIBHCMSG

AIBH0301E	The check found that the level of MLz Core is not at the recommended level for AISS 1.2.
------------------	---

Explanation

The check found that the installed Machine Learning for IBM z/OS Core Edition (MLz Core) is not at the recommended level for AI System Services for IBM z/OS 1.2.

System action

System processing continues.

Operator response

Notify the system programmer.

System programmer response

Consider upgrading to MLz Core 3.2, which is included with AI System Services for z/OS 1.2.

Module

AIBHCCHK, AIBHCMSG

z/OSMF messages

IZU messages

The following messages can be issued by z/OSMF while using the AI Control Interface for IBM z/OS.

IZUAM0001I	The training process that was initiated for <i>use-case</i> on <i>system</i> completed at <i>time</i>.
-------------------	---

Explanation

The training process successfully completed for the given use case on the given system.

In the message text:

use-case

The AI-enhanced use case that was performed.

system

The system on which the training process was performed.

time

The time when the training process completed.

System programmer response:

No action is required.

User response:

No action is required.

IZUAM0002W

number-of-component component-name of system was not trained due to insufficient data or data error.

Explanation

One or more components were skipped or failed during training.

In the message text:

number-of-component

The number of components that were skipped or have failed.

component-name

The general name of the components that are managed by AI.

system

The system on which the AI-powered use case was performed.

System programmer response:

Check whether the data is sufficient for training before you attempt to train again.

User response:

Check whether the data is sufficient for training before you attempt to train again.

IZUAM0020E

The z/OSMF server does not have the necessary permission to the AI Base Component of z/OS.

Explanation

The z/OSMF server ID does not have the correct permissions in order to call the AI Base Component for updating model update and checking status.

System programmer response:

Set up the correct SAF permission for the z/OSMF server ID to call the AI Base Component. Refer to the security sample job SYS1.SAMPLIB(IZUAISEC).

User response:

Contact your z/OSMF security administrator for help with setting up the correct authorization for the z/OSMF server ID to call the AI Base Component.

IZUAM0021E

An internal error occurred in the AI Base Component of z/OS on model model-name of use case use-case on system system-name.

Explanation

An internal error occurred in the AI Base Component of z/OS when trying to get the status of the given model on the given system.

In the message text:

model-name

The name of the model.

use-case

The name of the use case that provided the model.

system-name

The name of the system.

System programmer response:

No action is required.

User response:

Contact IBM support for diagnostics.

IZUAM0022E

The AI model of *model-name* for the *use-case-name* use case could not be found by the AI Base Component on system *system-name*.

Explanation

The requested AI model for the use case could not be found by the AI Base Component on the given system.

In the message text:

model-name

The name of the model.

use-case-name

The name of the use case that provided the model.

system-name

The name of the system.

System programmer response:

Check that the AI Base Component of z/OS is started and that the provider's AI model is registered.

User response:

Contact your system programmer.

IZUAM0023E

The AI Base Component of z/OS can't connect to the AI model server on system *system-name*.

Explanation

The AI Base Component of z/OS can't connect to the AI model server on the given system.

In the message text:

system-name

The name of the system.

System programmer response:

Check that the AI framework configuration is correct and the AI model server is running.

User response:

Contact your system programmer to make sure that the AI framework configuration is correct and the AI server is running.

IZUAM0024E

An error was detected in the *model-name* AI model of use case *use-case* on system *system-name* registered to the AI Base Component.

Explanation

An error was detected in the model by the AI Base Component during inferencing.

In the message text:

model-name

The name of the model.

use-case

The name of the use case that provided the model.

system-name

The name of the system.

System programmer response:

No action is required.

User response:

Contact IBM support for diagnostics.

IZUAM0025E**System *system-name* was not found.**

Explanation

The system could not be found by the AI Base component.

In the message text:

system-name

The name of the system.

System programmer response:

No action is required.

User response:

Make sure the system is running.

IZUAM0026E**The AI Base Component of z/OS has not started on system *system-name*.**

Explanation

The AI Base component has not started on the given system.

In the message text:

system-name

The name of the system.

System programmer response:

Make sure the AI Base component on the given system is running.

User response:

Contact your system programmer to make sure that the AI Base component on the given system is running.

IZUAM0027E**The AI Base Component of z/OS experienced an unexpected failure on system *system-name*.**

Explanation

The AI Base Component of z/OS experienced an unexpected failure on the given system.

In the message text:

system-name

The name of the system.

System programmer response:

Contact IBM support for diagnostics.

User response:

Contact IBM support for diagnostics.

IZUAM0100I**The status of *component-name* was not changed because its training status is *training-status*.**

Explanation

The status of the component was not changed because its training status is not qualified.

In the message text:

component-name

The name of the component.

training-status

The current training status of the component.

System programmer response:

No action is required.

User response:

Train the system before attempting to change the status of the component.

IZUAM0101I

The component *component-name* does not exist. Please refresh the current page.

Explanation

The component that you are trying to view does not exist.

In the message text:

component-name

The name of the component.

System programmer response:

No action is required.

User response:

Refresh the current page to view the components that are currently available.

IZUAM0102W

Data was not found for field *requested-data*.

Explanation

The requested data was not found.

In the message text:

requested-data

The requested data that was not found.

System programmer response:

Check whether the data request is correct.

User response:

Check whether the data request is correct.

IZUAM0103E

An error occurred while processing the request. Error: *error-details*.

Explanation

An error occurred while processing the request. The context of the error is provided.

In the message text:

error-details

The details of the error.

System programmer response:

Examine the logs for more information about this failure. Verify the setup of z/OSMF.

User response:

Contact your z/OSMF administrator or system programmer.

IZUAM0104I

No matching data was found with the search key in the data store *data-store-name*.

Explanation

There is no document found in the data store that matches the search key.

In the message text:

data-store-name

The name of the data store.

System programmer response:

Make sure the correct data store is provided and that the data collector is up and running.

User response:

Contact your system programmer.

IZUAM0105E	The use case <i>use-case-name</i> of provider <i>provider-name</i> was not found.
------------	---

Explanation

The requested provider or use case was not found.

In the message text:

use-case-name

The name of the AI-powered use case.

provider-name

The name of the provider.

System programmer response:

No action is required.

User response:

Refresh the current page to view the providers and use cases that are currently available.

IZUAM0106E	The system <i>system-name</i> was not defined in the use case <i>use-case-name</i> .
------------	--

Explanation

The requested system was not defined in the given use case.

In the message text:

system-name

The name of the system.

use-case-name

The name of the AI-powered use case.

System programmer response:

No action is required.

User response:

Refresh the current page to view the systems that are currently available.

IZUAM0107E	The UI configuration file <i>ui-config-file</i> on system <i>system-name</i> was not found.
------------	---

Explanation

The requested UI configuration file was not found on the given system.

In the message text:

ui-config-file

The path of the UI configuration file.

system-name

The name of the system.

System programmer response:

Make sure that the provided path for the UI configuration file is correct, and that the correct permissions are set for the z/OSMF server ID to read it.

User response:

Contact your system programmer.

IZUAM0108E	The UI configuration file <i>ui-config-file</i> on system <i>system-name</i> could not be read.
-------------------	--

Explanation

The z/OSMF server ID does not have the correct permissions to read the requested UI configuration file on the given system.

In the message text:

ui-config-file

The path of the UI configuration file.

system-name

The name of the system.

System programmer response:

Make sure that the provided path for the UI configuration file is correct, and that the correct permissions are set for the z/OSMF server ID to read it.

User response:

Contact your system programmer.

IZUAM0109E	The UI configuration file <i>ui-config-file</i> on system <i>system-name</i> could not be read. Error: <i>error-details</i>.
-------------------	---

Explanation

The requested UI configuration file on the given system could not be read. The context of the error is provided.

In the message text:

ui-config-file

The path of the UI configuration file.

system-name

The name of the system.

error-details

The details of the error.

System programmer response:

Correct the error to provide an available UI configuration file.

User response:

Contact your system programmer.

IZUAM0110E	An error occurred while retrieving the visualization definitions from the UI configuration file on system <i>system-name</i>. Error: <i>error-details</i>.
-------------------	---

Explanation

An error occurred while retrieving the visualization definitions from the UI configuration file on the given system. The context of the error is provided.

In the message text:

system-name

The name of the system.

error-details

The details of the error.

System programmer response:

Correct the error to provide an available UI configuration file.

User response:

Contact your system programmer.

IZUAM0111E	The component <i>component-name</i> on system <i>system-name</i> does not exist or is no longer in AI mode <i>AI-mode</i>.
-------------------	---

Explanation

The requested component does not exist or is no longer in the given AI mode.

In the message text:

component-name

The name of the component.

system-name

The name of the system.

AI-mode

The current state of AI operations on the component.

System programmer response:

No action is required.

User response:

Refresh the current page to view the components that are currently available.

IZUAM0112E	The data store was not defined in the use case <i>use-case-name</i> on system <i>system-name</i>.
-------------------	--

Explanation

The requested data store was not defined in the given use case.

In the message text:

use-case-name

The name of the AI-powered use case.

system-name

The name of the system.

System programmer response:

Make sure an available EzNoSQL data store is provided.

User response:

Contact your system programmer.

IZUAM0113E	An error occurred while connecting to the data store <i>data-store-name</i>. Error: <i>error-details</i>.
-------------------	--

Explanation

An error occurred while connecting to the requested data store. The context of the error is provided.

In the message text:

data-store-name

The name of the data store.

error-details

The details of the error.

System programmer response:

Correct the error to provide an available EzNoSQL data store.

User response:

Contact your system programmer.

IZUAM0114E	An error occurred while retrieving the data from the data store <i>data-store-name</i>. Error: <i>error-details</i>.
-------------------	---

Explanation

An error occurred while retrieving the data from the requested data store. The context of the error is provided.

In the message text:

data-store-name

The name of the data store.

error-details

The details of the error.

System programmer response:

Correct the error to provide an available EzNoSQL data store.

User response:

Contact your system programmer.

IZUAM0115E

No matching data was found for the component *component-name* in the data store *data-store-name*.

Explanation

The given data store does not contain any data for the requested component.

In the message text:

component-name

The name of the component.

data-store-name

The name of the data store.

System programmer response:

No action is required.

User response:

Contact your system programmer.

IZUAM0116W

No matching data was found in the given time range for the for the component *component-name* in the data store *data-store-name*. The earliest record of data for the component *component-name* is at *earliest-record-datetime*.

Explanation

The given data store does not contain any data for the component in the requested time range. The earliest record of data for the component is at the given date and time.

In the message text:

component-name

The name of the component.

data-store-name

The name of the data store.

earliest-record-datetime

The date and time of the earliest record of data.

System programmer response:

No action is required.

User response:

Contact your system programmer.

IZUAM0120E

The API user that was specified in the configuration workflow is not defined in the AI server, or the user certificate is invalid.

Explanation:

The API user that was specified in the configuration workflow is not defined in the AI server, or the user certificate is invalid.

System programmer response:

Define the API user that is specified in the configuration workflow of the AI server, or refer to the sample job SYS1.SAMPLIB(IZUAISEC) to verify that the certificate configuration is correct.

User response:

Contact your system programmer.

IZUAM0121E	The certificate information <i>cert-info</i> specified in the configuration workflow is not defined.
-------------------	---

Explanation

The certificate information that is specified in configuration workflow is not defined.

In the message text:

cert-info

The certificate information.

System programmer response:

Define a certificate in the configuration workflow.

User response:

Contact your system programmer.

IZUAM0122E	An error occurred in the AI server during the request to submit the model training job: <i>error-details</i>.
-------------------	--

Explanation

An internal error from the AI server occurred. The details of the error are provided.

In the message text:

error-details

The details of the error.

System programmer response:

Make sure the AI server is up and running, or contact IBM support for diagnostics.

User response:

Contact your system programmer to make sure the AI server is up and running, or contact IBM support for diagnostics.

IZUAM0123E	The process of system training failed because a callback was not received from the AI server.
-------------------	--

Explanation:

The process of system training failed because z/OSMF did not receive any callback from the AI server.

System programmer response

- Check if the AI server and z/OSMF server is running during the process of system training.
- Check the HTTP status code of the callback request in the training job log of the AI server. If the HTTP status code is 401, make sure that the functional user ID has the correct access to the following SAF profiles. You can use the z/OSMF Security Configuration Assistant to verify the security setup for the user.
 - READ APPL IZUDFLT
 - READ EJBROLE IZUDFLT.IzuManagementFacilityAIManager.izuUsers
 - READ ZMFAPLA IZUDFLT.ZOSMF.AIMANAGER

User response:

Contact your system programmer.

IZUAM0124E

An error occurred during the process of training the system *system-name*. Error details: *error-details*

Explanation

The process of system training cannot be completed because an error occurred.

In the message text:

system-name

The name of the system.

error-details

The details of the error.

System programmer response:

No action is required.

User response:

Contact IBM support for diagnostics.

IZUAM0125E

The process of system training failed because an error occurred: The field *field-name* was not found because the server was interrupted in the training process.

Explanation

The process of system training cannot be completed. The given field was not found because the server was interrupted in the training process.

In the message text:

field-name

The name of the field.

System programmer response:

No action is required.

User response:

Send the train system request again.

IZUAM0126E

A training process is in progress on the system. You cannot initiate training for an AI model until the current training process completes.

Explanation:

The system cannot be trained because there is already a system training in process. Only one training can happen at the same time.

System programmer response:

No action is required.

User response:

Wait for the current training process to finish, then send the request again.

IZUAM0127E

The training request cannot be completed because the configuration field *field-name* was not found.

Explanation

The training request cannot be completed because the configuration field was not found.

In the message text:

field-name

The name of the field.

System programmer response:

Correct the configuration step that contains the missing field.

User response:

Contact your system programmer to correct the configuration and try to send the request again.

IZUAM0128E**The configuration field *conf-name* of the AI server was not found.****Explanation**

The configuration field of the AI server is missing.

In the message text:

conf-name

The field name of the AI server configuration.

System programmer response:

Set the missing field in the configuration. Try to send the request again.

User response:

Contact your system programmer to set the missing field in the configuration. Try to send the request again.

IZUAM0131E**Training the component failed because an error occurred. Error details: *error-details*.****Explanation**

Training the component failed because an error occurred. The details of the error are provided.

In the message text:

error-details

The details for the error.

System programmer response:

Check whether the configuration of the model training application is correct, and that there is sufficient data for training.

User response:

Contact your system programmer.

IZUAM0132E**The AI server cannot connect: *error-details*.****Explanation**

The AI server cannot connect because an error occurred.

In the message text:

error-details

The details for the error.

System programmer response:

No action is required.

User response:

Check whether the AI server is up and running and if the configuration of the AI server is correct.

IZUAM0133E**A connection cannot be established to the AI server because *field-name parameter-name* cannot be loaded.****Explanation**

The certificate information cannot be loaded because an error occurred.

In the message text:

field-name

The name of the field.

parameter-name

The name of the parameter that is used to load the certificate.

System programmer response:

Make sure that the certificate of the API user that was configured in the workflow exists and is connected to the keyring.

User response:

Contact your system programmer.

MVS system codes

System completion codes

System completion code 291 is used to report abends within AI Base Component for IBM z/OS.

291**Explanation**

The AI Base Component for IBM z/OS encountered an error. A reason code is displayed along with the 291 abend code. The format of the abend code is *xxxx-rrrr*, where *rrrr* is the halfword reason code.

The reason codes are:

rrrr

An internal error in the AI Base Component for IBM z/OS.

Action: Search problem reporting databases for a fix to the problem. If no fix is found, contact the IBM Support Center. Provide the LOGREC data set and SVC dump, if one is issued.

System action

The system continues.

System programmer response

Search problem reporting databases for a fix for the problem. If no fix exists, contact the IBM Support Center.

Source

AI Base Component for IBM z/OS

Diagnosis information

ENQ/DEQ summary

AI Base Component for IBM z/OS adds the following entry to the summary of major and minor ENQ/DEQ names and resources listed in *z/OS MVS Diagnosis: Reference*.

Table 12. Summary of major and minor ENQ/DEQ names and resources		
Major (QNAME)	Minor (RNAME)	Resource - using modules
SYSZAIB	AI Base Component for IBM z/OS	Serializes the startup of the AI base component—AIBINIT.

Component trace

The following topics provide component trace information related to AI Framework for IBM z/OS. For complete information about component trace, see [z/OS MVS Diagnosis: Tools and Service Aids](#).

Table 13. Summary of BCP component traces that use the component trace service	
Component	Reference
AI Base Component	See "SYSAIB component trace."

Planning for component tracing

Table 14. Determining if a component has a parmlib member				
Trace	Parmlib member	Default member	Default tracing beginning at initialization	Sublevel traces
SYSAIB	CTIAIB00	CTIAIB00	Yes, options ENF, INIT, MISC, QUERY, RECOVERY, REG	No

Table 15. Component trace options				
Trace	Default size and size range	Size set by	Change size after IPL	Buffer location
SYSAIB	4 MB 16KB - 4 MB	CTIAIBxx member or REPLY for TRACE CT command	Yes, when restarting a trace after stopping it	In the AI Base Component for z/OS address space

Table 16. Trace request options for component traces	
Trace	Trace request OPTIONS parameter
SYSAIB	See "OPTIONS parameter" in "Requesting a SYSAIB trace".

Table 17. Location of trace buffers for components			
Component	Address space buffer	Data space buffer	Trace data set
SYSAIB	Yes	No	Yes

Obtaining a component trace

Table 18. How to request SVC dumps for component traces	
Trace	Request of SVC dump
SYSAIB	By DUMP or SLIP command

Viewing the component trace data

Table 19. Subcommands that format component trace records		
Trace	IPCS subcommand	CTTRACE OPTIONS parameter
SYSAIB	CTTRACE COMP(SYSAIB)	See "Formatting a SYSAIB trace."

SYSAIB component trace

The following table summarizes the information for requesting a SYSAIB component trace for AI Base Component for IBM z/OS.

Information	For SYSAIB
Parmlib member	CTIAIB00. Default and only member: CTIAIB00 If no valid CTIAIB00 member exists, default tracing with options ENF, INIT, MISC, QUERY, RECOVERY and REG is activated.
Default tracing	Yes, entries for options entries for options ENF, INIT, MISC, QUERY, RECOVERY and REG
Trace request OPTIONS parameter	In CTIAIB00 member or REPLY for TRACE command
Buffer	<ul style="list-style-type: none">• Default: 4M• Range: 16K–4M• Size set by: CTIAIB00 member or REPLY to TRACE CT command• Change size after IPL: Yes, when restarting the AI Base Component for z/OS• Location: In the AI Base Component for z/OS address space
Trace records location	Address space buffer
Request of SVC dump	By DUMP or SLIP command
Trace formatting by IPCS	CTRACE COMP(SYSAIB)
Trace format OPTIONS parameter	Yes

Requesting a SYSAIB trace

Specify options for requesting a SYSAIB component trace in a CTIAIB00 parmlib member or on the REPLY for a TRACE CT command.

You can change the options for SYSAIB tracing while the trace is running.

CTIAIB00 parmlib member

The following table indicates the parameters you can specify in a CTIAIB00 parmlib member.

Parameters	Allowed in CTIAIB00 member
ON or OFF	One is required.
ASID	No
JOBNAME	No
BUFSIZE	Yes
OPTIONS	Yes
SUB	No
PRESET	No
LIKEHEAD	No
WTR	Yes

Parameters	Allowed in CTIAIB00 member
WTRSTART or WTRSTOP	Yes

IBM supplies the CTIAIB00 parmlib member, which specifies the tracing for AI Base Component for IBM z/OS that begins when the AIBPROC procedure is started. The contents of the CTIAIB00 member are:

```
TRACEOPTS
ON
OPTIONS('ENF','INIT','MISC','QUERY','RECOVERY','REG')
BUFSIZE(4M)
```

TRACE and REPLY commands

The following tables indicate the parameters you can specify on TRACE CT commands and a REPLY.

Parameters	Allowed on TRACE CT for trace?
ON, <i>nnnnK</i> , <i>nnnnM</i> , or OFF	One is required.
COMP	Required
SUB	No
PARM	Yes

Parameters	Allowed on TRACE CT for writer?
WTRSTART or WTRSTOP	One is required if a writer is being used.

Parameters	Allowed on REPLY for trace?
ASID	No
JOBNAME	No
OPTIONS	Yes
WTR	Yes

You can change options while a SYSAIB trace is running. However, to change the buffer size, you must stop the trace and restart it with the new buffer size.

OPTIONS parameter

The following values are valid for the OPTIONS parameter for the CTIAIB00 parmlib member and the reply for a TRACE command:

ALL

Traces all events for AI Base Component for z/OS.

ENF

Traces events for ENF usage.

INF

Traces events for inference processing (AIBXINF).

INIT

Traces events about the initialization of AI Base Component for z/OS.

MINIMUM

Traces a minimal set of events for AI Base Component for z/OS.

MISC

Traces miscellaneous events for AI Base Component for z/OS.

QUERY

Traces events about querying model status (AIBXQRY).

RECOVERY

Traces recovery events in AI Base Component for z/OS.

REG

Traces events for registering the instance of a model (AIBXREG, AIBXDREG).

Examples of requesting a SYSAIB trace

- Example CTIAIB00 member:

This member requests ALL AI Base Component for z/OS tracing:

```
TRACEOPTS
ON
OPTIONS('ALL')
BUFSIZE(4M)
```

- Example TRACE command:

This command requests a trace of INIT and QUERY trace events:

```
TRACE CT,ON,COMP=SYSAIB
42 ITT006A ...
REPLY 42,OPTIONS=(INIT,QUERY),END
```

Formatting a SYSAIB trace

Use the following IPCS subcommand to format a SYSAIB trace:

```
CTRACE COMP(SYSAIB) FULL
```

Output from a SYSAIB trace

The following example shows sample output from a SYSAIB trace:

```
SY1      REG      00000001 19:22:28.368997 Registration
ASID..001A  MODNAME..AIBPREG  TCB..004F8C50  STACK..7EFF8F28
Return Code = 00000000
Reason Code = 00000000
Reg Token = 00000050 00000040 00000008 00000001
Provider Name = SYSAIUSER
Use Case Name = AIUSECASE
Model Name = AIMODEL
```

SVC dump directory

AI Base Component for IBM z/OS adds the following entries to the SVC dump directory in z/OS MVS *Diagnosis: Reference*.

COMPON=AIB ENF BROADCAST,COMPID=SCAIB,ISSUER=AIBPAIM

Component

AI Base Component for IBM z/OS (5752-SCAIB)

Issuing module

AIBPAIM

Explanation

An error occurred during AIBPAIM processing.

**COMPON=AIB,
COMPID=SCAIB,ISSUER=AIBMIREC,MODULE=aaaaaaaa+bbbb,ABEND=cccc
c,REASON=dddddddd**

Component

AI Base Component for IBM z/OS (5752-SCAIB)

Issuing module

AIBMIREC

Explanation

An error occurred within an AI Base Component for z/OS module. In the dump title, the variables are:

aaaaaaaa

The 8-character module name that encountered the error.

bbbb

The offset within the module where the error occurred. A value of ???? indicates that the offset could not be determined.

cccc

The abend code.

dddddddd

The reason code.

Health checks

AI Base Component for IBM z/OS checks (IBMAIB)

AIB_AISS_LEVEL

Description:

Checks that the installed version of Machine Learning for IBM z/OS Core Edition (MLz Core) is at the recommended level for AI System Services for IBM z/OS 1.2.

Reason for check:

Running with an older version of MLz Core lacks functionality and protection that is available at later releases.

z/OS release the check applies to:

z/OS 3.1 and later.

Parameters accepted:

None.

User override of IBM values:

The following sample shows the defaults for customizable values for this check. Use this sample to make permanent check customizations in an HZSPRMxx parmlib member used at IBM Health Checker for z/OS startup. If you just want a one-time only update to the check defaults, omit the first line (ADDREPLACE POLICY) and use the UPDATE statement on a MODIFY *hzsproc* command. Note that using non-POLICY UPDATES in HZSPRMxx can lead to unexpected results and is, therefore, not recommended.

You can copy and modify the following example to override the check defaults:

```
ADDREPLACE POLICY[(polycname)] [STATEMENT(name)]  
UPDATE  
CHECK(IBMAIB,AIB_AISS_LEVEL),  
INTERVAL(ONETIME),  
SEVERITY(LOW),  
DATE('date_of_the_change'),  
REASON('Your reason for making the update')
```

Reference:

For additional information, see [Infusing AI into IBM z/OS](#).

Messages:

This check issues the following messages:

- AIBH0300I
- AIBH0301E

See [AIB messages](#) in *z/OS MVS System Messages, Vol 1 (ABA-AOM)*.

SECLABEL recommended for MLS users:

SYSLOW. See [z/OS Planning for Multilevel Security and the Common Criteria](#) for information on using SECLABELs.

Part 3. AI providers and use cases

This part presents information about the providers that utilize the AI Framework for IBM z/OS and their use cases for AI-enhanced system management.

- AI-powered WLM batch initiator management

Chapter 10. AI-powered WLM batch initiator management

AI-powered WLM batch initiator management uses artificial intelligence (AI) to predict workload spikes and proactively adjust the number of batch initiators to meet the demand.

Batch workload management

Job initiators for WLM-managed job classes are dynamically controlled by workload management (WLM). WLM adjusts the number of initiators on each system based on the following factors:

- The queue of jobs awaiting execution in WLM-managed job classes
- The performance goals and relative importance of this work
- The success of meeting these goals
- The capacity of each system to do more work

Batch jobs can be regularly scheduled or submitted in an ad-hoc manner or both, so at any time there is a combination of regular and random workloads running.

To allow batch jobs to run, WLM starts and stops initiators. The number of active initiators determines the number of jobs that can run in parallel, per system and service class.

Figure 2 on page 79 illustrates the organization of batch workloads by service classes that run on WLM-managed initiators.

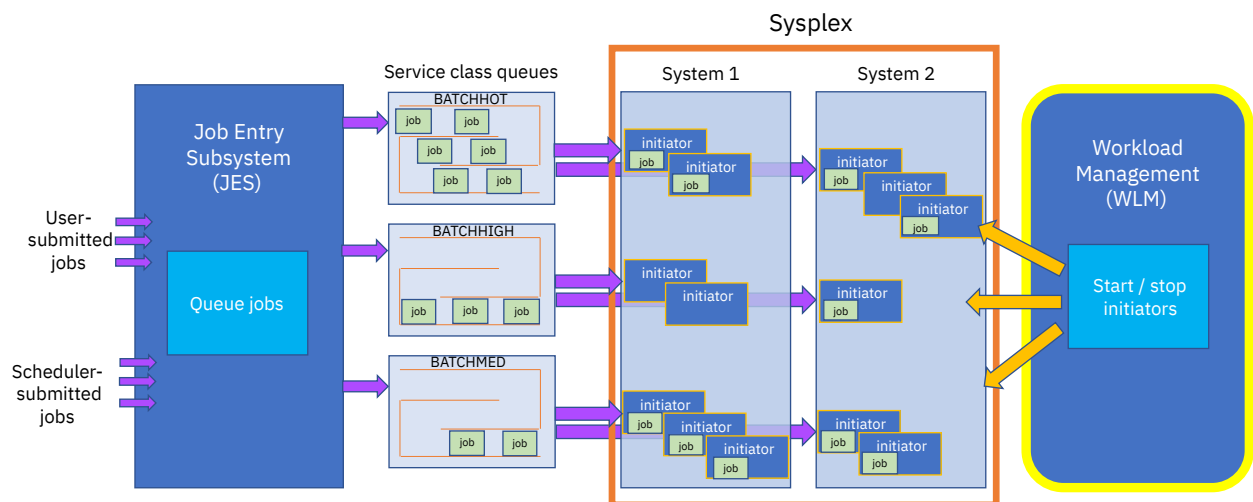


Figure 2. z/OS batch job assignment by service classes

While batch workload management by WLM is very efficient, it is primarily a *reactive* system. In other words, WLM reacts to workloads as they arrive, which is sufficient for most purposes. One consequence, however, is that if many jobs are regularly added to the input queue for a given service class, it can take some time for WLM to start initiators, if the service definition and system load allow.

AI-powered batch workload management

With z/OS 3.1, WLM introduces AI to its internal processing to predict any regular spikes by analyzing and modelling historically repeating batch workloads. For instance, regular spikes in batch jobs might typically occur every few minutes, at certain times of each weekday, or only at end of month. Such workload behaviors can vary between day and night, and between different days of the week, and each service class can have its own workload patterns.

The ability to use AI to predict such regular workload patterns allows WLM to start initiators for each service class *before* the next regular workload spike occurs. This *proactive* management of initiators allows workloads to start and complete sooner and keeps queue lengths to a minimum. In other words, each service class is better able to achieve the goals defined in the WLM policy.

Figure 3 on page 80 shows how WLM's management of initiators could differ between batch workload management and AI-powered batch workload management.

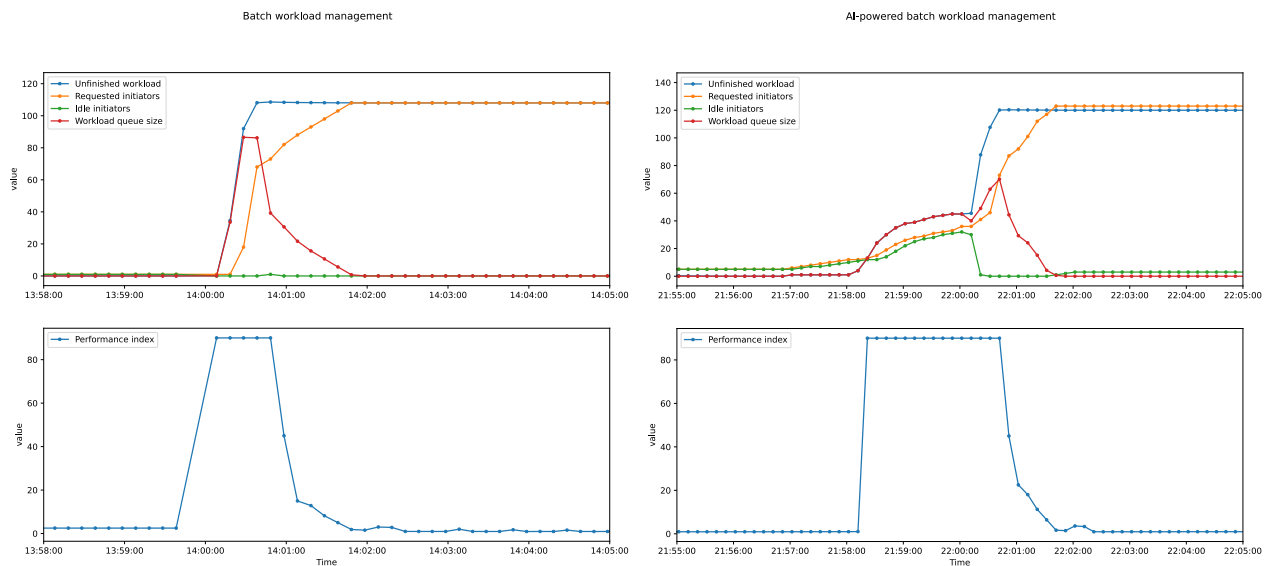


Figure 3. Workload and initiator behavior for batch workload management vs. AI-powered batch workload management

In the figure:

- With batch workload management, the system reacts to an observed workload spike and starts initiators after the workload arrives—the workload curve rises before the initiator curve. The workload queue size curve also rises as jobs must wait until enough initiators have been started.
- With AI-powered batch workload management, the system predicts the workload spike and proactively starts initiators before the workload arrives—the initiator curve begins to rise before the workload curve and continues to rise until enough initiators have been started. In this way, jobs can start running sooner and the workload queue size remains smaller.

AI in WLM

As described earlier in "AI framework components," the AI processing in z/OS is internal to the operating system and WLM. Neither system administrators nor users require any skills in data science or machine learning because z/OS provides a framework for AI which hides this complexity.

As an AI *provider* that uses the z/OS AI Framework, WLM provides the training application that is used to create and train the AI model for the AI-powered WLM batch initiator management use case.

Configuring and managing the AI-powered WLM batch initiator management use case

Before you can use this WLM use case, you must use the z/OSMF-based AI Framework for IBM z/OS Configuration Workflow and the Configuring AI-Powered Initiator Management for WLM Workflow to configure the components that make up the z/OS AI Framework and this AI use case, as described earlier in "Configuring the z/OS AI Framework."

Configuration notes for this use case:

- The z/OSMF workflow to configure this WLM use case is separate from the main AI Framework configuration workflow. Be sure you have successfully completed the AI Framework for IBM z/OS

Configuration Workflow (or the Configure AI Framework for IBM z/OS for Additional Systems Workflow) before attempting to use the Configuring AI-Powered Initiator Management for WLM Workflow to configure this use case.

- Before you can use the Configuring AI-Powered Initiator Management for WLM Workflow, you must create an instance of the workflow from its workflow definition file, just as you did for the main AI Framework configuration workflow, as described earlier in "Configuring the z/OS AI Framework." The process to create and use the Configuring AI-Powered Initiator Management for WLM Workflow is similar to what you have already done for the main AI Framework configuration workflow.
- Substitute the following information into the procedure described earlier in "Configuring the z/OS AI Framework" to create and use the Configuring AI-Powered Initiator Management for WLM Workflow.
 - On the **Create Workflow** dialog, enter the following path in the **Workflow definition file** field:

```
/usr/lpp/bcp/wlm/ainitiator/workflow/WLM_Config.xml
```

- In the **Workflow variable input file** field, enter the path of the property file into which you exported variables from your initial configuration using the main AI Framework configuration workflow. This allows the workflow to import those variable values to prime the workflow for the WLM use case.



Attention: Do not omit this step. Otherwise, the WLM configuration workflow will not have all the information it expects to receive from the main AI Framework configuration workflow and will fail on the step that validates the imported variables.

- After you create an instance of the Configuring AI-Powered Initiator Management for WLM Workflow, you can begin the workflow and perform the steps just as you did for the main AI Framework configuration workflow.

After you successfully configure the z/OS AI Framework and this AI use case, you can use the AI Control Interface for IBM z/OS to manage the batch service classes for this WLM use case, as described in [“Controlling and managing AI-powered WLM batch initiator management” on page 82](#). You can find the **AI Control Interface** task on the z/OSMF desktop.

About the AI model

AI models are not delivered as part of the z/OS AI Framework installation. The *training application* is used to create and train system-specific machine learning models based on historical SMF type 99 subtype 2 data from that system. These so-called *time series models* recognize regular historical patterns in the data pertaining to WLM-managed batch initiators and, thus, can predict upcoming regular changes in each system’s workload. Such regular patterns often result from the use of a scheduling mechanism to introduce batch jobs into systems, and patterns can recur over minutes, hours, or even days. The resulting pattern recognition allows each model to provide WLM with a continuous heads-up a few minutes before any workload changes, as shown in the example in [Figure 3 on page 80](#). For instance, if a regular spike in batch initiators can be expected at 11:00, the model will start predicting this increase a few minutes before 11:00, which allows WLM to make earlier decisions and, if desirable, to have more initiators already started when a workload surge arrives.

Note: Regardless of the model's prediction, the final decision on whether to start initiators always rests with WLM.

The performance of this type of AI model depends on the similarity of the training data to the current system behavior. The largest benefit from AI will be for service classes that have a high regularity in workload behavior. Turning on AI for service classes that have a random workload behavior is unlikely to provide benefit. The regular behavior of a system can vary over time, so it is advisable to retrain models at least every 4 weeks using the most current SMF data. If the workload behavior is known to have significantly changed recently, training can be performed earlier. Training is performed by using the AI Control Interface for z/OS.

How WLM uses AI

When AI is enabled for a service class, there are 3 basic scenarios that describe how WLM might use the predictions that it receives from the AI model and the resulting behavior:

- **Scenario 1:** The model predicts a need to start initiators to handle a workload spike, and the workload arrives, as expected.

Result: WLM starts initiators before the workload arrives, if it is prudent to do so based on other conditions in the system at that time. For instance, WLM might not start initiators if the system is already heavily loaded, or there is other work with higher importance, or the service class is associated with a capped resource group.

- **Scenario 2:** The model predicts a need to start initiators, but no workload spike actually occurs.

Result: WLM starts initiators, if prudent to do so, but if no workload arrives. WLM then stops the idle initiators.

- **Scenario 3:** The model provides no predicted need to start initiators, but a workload spike occurs.

Result: WLM manages the workload as it normally would without AI.

When AI is in simulation mode for a service class, AI still provides predictions to WLM, but WLM does not use them to manage the service class. WLM manages the service class without AI, as it always has. Simulation mode allows you to see how the model would have performed if AI were enabled for the service class.

Controlling and managing AI-powered WLM batch initiator management

You can control whether or not WLM uses AI for batch workload management by using an OPT parameter in parmlib and by using the z/OSMF-based AI Control Interface for IBM z/OS.

Using OPT to control AI management in WLM

AI management within WLM is enabled by default.

To deactivate AI management, WLM provides the **AIMANAGEMENT** parameter in the IEAOPTxx member of parmlib:

```
AIMANAGEMENT={YES|NO}
```

Setting AIMANAGEMENT=NO deactivates all use of AI in WLM. The default is AIMANAGEMENT=YES.

However, understand that WLM only uses AI for batch workload management when all of the following conditions are true:

- The **AIMANAGEMENT** parameter is not set to NO in the active IEAOPTxx member—that is, the parameter is either explicitly set to YES or allowed to default to YES.
- All components of the AI Framework for z/OS are installed, configured, and started.
- The AI Control Interface for z/OS shows a **Train status** of Trained and an **AI mode** of Enabled for a given service class. (See [Figure 4 on page 83](#).)

Using the AI Control Interface for z/OS to manage AI-powered WLM batch initiator management

After the AI Framework for z/OS has been successfully configured, you can use the AI Control Interface for z/OS to choose which service classes associated with batch workload management you would like to enhance with AI, and train the system.

Training requires that you have at least 30 days of contiguous, historical SMF type 99 subtype 2 data available. The z/OS AI Interface indicates whether sufficient SMF data is available for training. Training

is done at the system level for all service classes; however, you can decide to enable or disable AI for individual service classes or to place a service class into AI simulation mode.

Figure 4 on page 83 shows an example of the **Manage - Components** view in the z/OS AI Interface after training has completed on a particular system. For this use case, the components are the WLM service classes related to batch work.

Note: In order for a batch service class to appear in the list of service classes, there must be some job activity (that is, at least one job must be run) for that service class.

Service class (batch work)	Training status	AI performance metrics	AI mode
AIBATCH1	Trained	Review results	Simulating
AIBATCH2	Trained	Review results	Simulating
AIBATCH3	Trained	Review results	Simulating
AIBATCH4	Trained	Not applicable	Enabled
AIBATCH5	Trained	Not available	Disabled

Figure 4. The AI Control Interface for IBM z/OS: Manage - Components view

As shown in the figure, AI is enabled on a per-system basis for specific service classes. In the figure, there are 5 service classes that are available for AI-powered WLM batch initiator management on system N67 and all have been trained:

- AIBATCH4 is actively using AI for batch initiator management, as indicated by Enabled in the **AI mode** column. In this mode, AI is running and advising WLM, and WLM is using the advice to manage the initiators in this service class. The system must be trained on historical data before this mode can be used.
- AIBATCH 1, AIBATCH2, and AIBATCH3 have an AI mode of Simulating. In this mode, AI is running and advising WLM, but WLM is *not* using the advice to manage the initiators in these service classes; WLM continues to manage these service classes normally, as if AI were not enabled. However, the AI predictions are being recorded for observation, which you can see then you click **Review results** in the **Performance metrics** column. The performance metrics provide a graphical display that allows you to compare over time the number of initiators predicted by AI to the number of active initiators that WLM is using. You can drill down on the display to focus on specific time intervals. The system must be trained on historical data before this mode can be used.
- AIBATCH5 is not using AI for batch initiator management, as indicated by Disabled in the **AI mode** column. In this mode, AI is not running nor advising WLM; WLM manages the service class normally, without AI. You can change the AI mode to Enabled or Simulating later, if desired, after successful training and the **Training status** column indicates Trained.

Within the z/OS AI Interface, the information icons beside each column heading display tool tips. Further information is also available by clicking the help icon in the title bar. Also refer to the topic, "Using the AI Control Interface for z/OS," earlier in this documentation.

Reference information related to the AI-powered WLM use case

PARMLIB

IEAOPTxx (OPT parameters)

The OPT parameters allow the installation to change many system resources manager (SRM) constants. The following list categorizes the options as special including, for example, abnormal termination or whether non-enclave transaction work of queue servers and enclave servers is to be managed or not, adjusting options for constants, or options for special assist processors.

The OPT contains several categories of information. The keywords do not have to be grouped by category and can appear in any order. The system ignores both repetition of a keyword and the use of keywords that were used in previous releases and are no longer supported.

Special options

- [,ABNORMALTERM=*option*]
- [,ABSMSUCAPPING=*option*]
- [,AIMANAGEMENT=YES|NO]
- [,CNTCLIST=*option*]
- [,DVIO=*option*]
- [,FULLPRESYSTEM=YES|NO]
- [,HIPERDISPATCH=YES|NO]
- [,INITIMP=*option*]
- [,MANAGENONENCLAVEWORK=YES|NO]
- [,VARYCPU=*option*]
- [,VARYCPUMIN=*nn*]
- [,STORAGESERVERMGT=YES|NO]
- [,SUPPSAFINFOMSG=YES|NO]

Statements and parameters for IEAOPTxx

AIMANAGEMENT=YES|NO

Specifies whether WLM is to use artificial intelligence (AI) for WLM-managed batch initiators for all service classes that are enabled for AI.

WLM only uses AI for batch initiator management when all of the following conditions are true:

- The **AIMANAGEMENT** parameter in the active IEAOPTxx member is set to (or defaulted to) YES.
- The AI Framework for IBM z/OS components are configured and started, including the AI model server and the AI Base Component for IBM z/OS.
- The AI model for WLM batch initiator management is trained and deployed, and enabled for specific service classes via the AI Control Interface for IBM z/OS task in z/OSMF.

Event notification facility

Event code 56 has a new WLMENF56_QUAL_QUEUE_REG qualifier in support of the AI-powered WLM batch initiator management use case.

The following characteristics vary depending on the event for which you are listening.

Event code

Identifies the event.

Qualifier

Further defines the specific event for which you would like to listen.

Parameter list

Passes information about the event to the listener user exit.

Exit type

Specifies the type of the listener user exit routine, which can be either EXIT or SRBEXIT.

Cross-system capable

Specifies whether the exit is to receive signals from other systems in the sysplex

Table 20 on page 85 describes the ENF event codes.

Table 20. ENF macro event codes				
Event code	Description	Qualifier	Parameter list passed to the user exit	Exit type / Cross-system capable
56	Workload management has changed an attribute of a job.	WLMENF56_QUAL_RESET A job was reset using the RESET system command or IWMRESET macro. WLMENF56_QUAL_ENCLAVERESET An enclave has been successfully reset via the IWMERES service. WLMENF56_QUAL_QUEUE_REG A batch queue for a service class is to be registered using the IWMBREG service.	Mapped by IWMREN2	EXIT / NO

SMF records**Record type 72 subtype 3 – Workload activity**

The z/OS Data Gatherer writes SMF type 72 subtype 3 records for each service class and active report class in the active service policy.

For complete details about the type 72 record, see [z/OS MVS System Management Facilities \(SMF\)](#).

Workload manager control section

Offsets	Name	Length	Format	Description
:				

Offsets	Name	Length	Format	Description
1 1	R723MFLG	1	binary	Flags. Bit Meaning when set 0 Indicator for zAAP crossover. 1 Indicator for zAAP honor priority. 2 Indicator for zIIP honor priority. 3 Failure returned by HISMT service. Multithreading maximum capacity numerator values are invalid. 4 Indicator that service class is not eligible for honor priority processing. When on, specialty engine eligible work in this service class will not be offloaded to CPs for help processing. 5 Indicator for a tenant report class 6 Service class and tenant report class periods that are associated with a resource group and have assigned a discretionary goal are excluded from workload management. 7 WLM batch initiator management is AI-infused.
:				

Service/report class period data section

Offsets	Name	Length	Format	Description
:				
11 B *		1	binary	Reserved.
:				

Record type 99 subtype 2 – Service class data

SMF type 99 subtype 2 records contain data for service classes. A subtype 2 record is written every policy interval for each service class if any period in the service class had recent activity.

For complete details about the type 99 subtype 2 record, see [z/OS MVS System Management Facilities \(SMF\)](#).

Period data section

Offsets	Name	Length	Format	Description
:				
776 308	SMF99_AI_OF	4	binary	Offset to AI data section from beginning of record (including RDW) .
780 30C	SMF99_AI_LN	2	binary	Length of AI data entry.
782 30E	SMF99_AI_ON	2	binary	Number of AI data entries.

AI data section

The AI data section is present only for periods participating in WLM AI batch initiator management.

Triplet information: This section is located on the record using the following triplet fields, which are located in "Period data section".

Offset

SMF99_AI_OF

Length

SMF99_AI_LN

Number

SMF99_AI_ON

The offset values shown are relative to the beginning of the SMF99_S2_AI_DATA_MAP section.

Offsets	Name	Length	Format	Description
0	0 SMF99_S2_AI_DATA_MAP	112	*	SMF99 subtype 2 AI data section.
0	0 SMF99_S2_AI_MODELNAME	32	EBCDIC	Model name identifier.
32	20 SMF99_S2_AI_MODELVERSION	12	EBCDIC	Model version.
44	2C SMF99_S2_AI_MODELUSECASE	16	EBCDIC	Model use case identifier
60	3C SMF99_S2_AI_FLAGS	4	binary	AI flags: <div> Bit Meaning when set 0 AI predictions are enabled for this service class. 1 AI predictions are in simulation mode for this service class. 2–31 Reserved. </div>
64	40 SMF99_S2_AI_INFTIME	8	binary	Duration of last inference request in TOD format.
72	48 SMF99_S2_AI_DATA0	8	binary	Time stamp of last inference result in TOD format.
80	50 SMF99_S2_AI_DATA1	4	binary	AI data: Active server prediction.
84	54 SMF99_S2_AI_DATA2	4	binary	AI data: Active server prediction error.
88	58 SMF99_S2_AI_DATA3	4	binary	AI data: CP service prediction.
92	5C SMF99_S2_AI_DATA4	4	binary	AI data: CP service prediction error.
96	60 SMF99_S2_AI_DATA5	4	binary	AI data: zIIP service prediction.
100	64 SMF99_S2_AI_DATA6	4	binary	AI data: zIIP service prediction error.
104	68 SMF99_S2_AI_DATA7	4	binary	AI data: Model data fetch time.
108	6C SMF99_S2_AI_DATA8	4	binary	AI data: Model processing time.

Version information

When evaluating an SMF type 99 record, if the record version number in SMF99RVN is at least 33, the triplet fields for the AI data section exist in the record. A program can compare the value in SMF99RVN against the equate named SMF99_VER33 to verify that the fields exist in the record.

If the triplet fields exist, the value in SMF99_AI_ON can be used to determine if AI data sections are present.

Monitor III measurement tables

The following z/OS Data Gatherer measurement tables contain data related to AI-powered WLM batch initiator management. For complete details about z/OS Data Gatherer measurement tables, see [z/OS Data Gatherer Programmer's Guide](#).

ERBASIG3 – Address space identification table

Dec offset	Hex offset	Name	Length	Format	Description
⋮					
324	144	ASIMSTS	4	binary	<p>Miscellaneous states.</p> <p>Bit</p> <p>Meaning when set</p> <p>0 Address space is OMVS related</p> <p>1 Address space matched a classification rule in the active policy which prevents managing the region based on the response time goals of its served transactions</p> <p>2 CPU protection was assigned either to the address space or to transaction service classes being served by the space, and SRM is honoring the protection</p> <p>3 Storage protection was assigned either to the address space or to transaction service classes being served by the space, and SRM is honoring the protection</p> <p>4 This address space provides service to transactions classified to a different class than the address space itself</p> <p>5 WLM is managing this address space to meet the goals of work in other service classes</p> <p>6 Address space is a CICS TOR that matched a classification rule in the active policy which allows managing the region based on the region goals but also ensures that completed transactions are reported and used for management of the CICS AORs</p> <p>7 I/O priority group HIGH was assigned either to the address space or to transaction service classes served by the address space</p> <p>8 Address space is currently associated with a tenant resource group.</p> <p>9 Reserved.</p> <p>10 Address space matched a classification rule in the active policy which enables for recovery process boost.</p> <p>11 CPU protection was implicitly assigned.</p> <p>12–31 Reserved.</p>
⋮					

ERBRCDG3 – Resource collection data

Resource collection data entry:

Dec offset	Hex offset	Name	Length	Format	Description
⋮					

Dec offset	Hex offset	Name	Length	Format	Description
20	14	RCDSRVFLG	2	binary	Flags for service classes. Bit Meaning when set 0 WLM batch initiator management is AI-infused. 1–15 Reserved.
⋮					

Resource collection data - period entry:

Dec offset	Hex offset	Name	Length	Format	Description
⋮					
1	1	RCDPFLG1	1	binary	Service and Report class period flags Bit Meaning when set 0 Heterogeneous report class period. 1 Report class data accumulated during the interval. 2 Service class period implicitly designated CPU critical. 3–7 Reserved
⋮					

IWM messages

The following messages are issued by MVS Workload Management (WLM).

IWM054I **FAILURE IN *wlm-component*, PROCESSING DISABLED**

Explanation

For the specified *wlm-component*, processing has been disabled due to an unrecoverable error.

This message is issued to indicate that a function (*wlm-component*) of WLM is now unavailable. The MVS system continues to run in a degraded fashion. An IPL is required to enable the function.

The following functions may become unavailable:

wlm-component

ARM COMPLETIONS RECORDING

WLM support for the guest platform management provider that tracks completions of Application Response Measurement (ARM) instrumented processes.

LPAR CPU MANAGEMENT

For systems in LPAR mode, WLM support for managing LPAR weights and controlling the optimal number of CPUs is disabled.

DYNAMIC CHANNEL PATH MANAGEMENT

WLM support for managing the number of channels through use of dynamically managed channel paths is disabled.

WLM RT TASK

WLM support for switching the MT mode of processors is disabled.

WLM AI TASK

WLM support for AI-powered batch workload management is disabled.

System action

None

System programmer response

Determine how long the system can continue without the WLM function and plan when a reIPL of MVS will cause the *wlm-component* to become available again.

Source

Workload management (WLM)

Module

IWML4WSA

Routing code

2,7

Descriptor code

M2

IWM091I	WLM AI MODEL <i>model-name</i> EXCEEDED FAILURE THRESHOLD. MODEL WILL BE DEREGISTERED.
----------------	---

Explanation

WLM encountered repeated errors with the specified AI model.

System action

WLM will deregister the model.

Operator response

None.

System programmer response

Look for AIB messages for more information. If you want to register the model again, restart WLM AI management by issuing the SET OPT=xx command to activate an IEAOPTxx member with the AIMANAGEMENT=NO parameter. Then, activate this member again with the AIMANAGEMENT=YES parameter. If the issue persists after restarting AI management, contact IBM Support.

Source

Workload management (WLM)

Module

IWMA9TSK

Routing code

2, 10

Descriptor code

4

Accessibility

Accessible publications for this product are offered through [IBM Documentation for z/OS \(www.ibm.com/docs/en/zos\)](http://www.ibm.com/docs/en/zos).

If you experience difficulty with the accessibility of any z/OS documentation see [How to Send Feedback to IBM](#) to leave documentation feedback.

Notices

This information was developed for products and services that are offered in the USA or elsewhere.

IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not grant you any license to these patents. You can send license inquiries, in writing, to:

*IBM Director of Licensing
IBM Corporation
North Castle Drive, MD-NC119
Armonk, NY 10504-1785
United States of America*

For license inquiries regarding double-byte character set (DBCS) information, contact the IBM Intellectual Property Department in your country or send inquiries, in writing, to:

*Intellectual Property Licensing
Legal and Intellectual Property Law
IBM Japan Ltd.
19-21, Nihonbashi-Hakozakicho, Chuo-ku
Tokyo 103-8510, Japan*

The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law: INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

This information could include missing, incorrect, or broken hyperlinks. Hyperlinks are maintained in only the HTML plug-in output for IBM Documentation. Use of hyperlinks in other output formats of this information is at your own risk.

Any references in this information to non-IBM websites are provided for convenience only and do not in any manner serve as an endorsement of those websites. The materials at those websites are not part of the materials for this IBM product and use of those websites is at your own risk.

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact:

*IBM Corporation
Site Counsel
2455 South Road*

Poughkeepsie, NY 12601-5400
USA

Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

The licensed program described in this document and all licensed material available for it are provided by IBM under terms of the IBM Customer Agreement, IBM International Program License Agreement or any equivalent agreement between us.

Any performance data contained herein was determined in a controlled environment. Therefore, the results obtained in other operating environments may vary significantly. Some measurements may have been made on development-level systems and there is no guarantee that these measurements will be the same on generally available systems. Furthermore, some measurements may have been estimated through extrapolation. Actual results may vary. Users of this document should verify the applicable data for their specific environment.

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 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.

All statements regarding IBM's future direction or intent are subject to change or withdrawal without notice, and represent goals and objectives only.

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental.

COPYRIGHT LICENSE:

This information contains sample application programs in source language, which illustrate programming techniques on various operating platforms. You may copy, modify, and distribute these sample programs in any form without payment to IBM, for the purposes of developing, using, marketing or distributing application programs conforming to the application programming interface for the operating platform for which the sample programs are written. These examples have not been thoroughly tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs. The sample programs are provided "AS IS", without warranty of any kind. IBM shall not be liable for any damages arising out of your use of the sample programs.

Terms and conditions for product documentation

Permissions for the use of these publications are granted subject to the following terms and conditions.

Applicability

These terms and conditions are in addition to any terms of use for the IBM website.

Personal use

You may reproduce these publications for your personal, noncommercial use provided that all proprietary notices are preserved. You may not distribute, display or make derivative work of these publications, or any portion thereof, without the express consent of IBM.

Commercial use

You may reproduce, distribute and display these publications solely within your enterprise provided that all proprietary notices are preserved. You may not make derivative works of these publications, or

reproduce, distribute or display these publications or any portion thereof outside your enterprise, without the express consent of IBM.

Rights

Except as expressly granted in this permission, no other permissions, licenses or rights are granted, either express or implied, to the publications or any information, data, software or other intellectual property contained therein.

IBM reserves the right to withdraw the permissions granted herein whenever, in its discretion, the use of the publications is detrimental to its interest or, as determined by IBM, the above instructions are not being properly followed.

You may not download, export or re-export this information except in full compliance with all applicable laws and regulations, including all United States export laws and regulations.

IBM MAKES NO GUARANTEE ABOUT THE CONTENT OF THESE PUBLICATIONS. THE PUBLICATIONS ARE PROVIDED "AS-IS" AND WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING BUT NOT LIMITED TO IMPLIED WARRANTIES OF MERCHANTABILITY, NON-INFRINGEMENT, AND FITNESS FOR A PARTICULAR PURPOSE.

IBM Online Privacy Statement

IBM Software products, including software as a service solutions, ("Software Offerings") may use cookies or other technologies to collect product usage information, to help improve the end user experience, to tailor interactions with the end user, or for other purposes. In many cases no personally identifiable information is collected by the Software Offerings. Some of our Software Offerings can help enable you to collect personally identifiable information. If this Software Offering uses cookies to collect personally identifiable information, specific information about this offering's use of cookies is set forth below.

Depending upon the configurations deployed, this Software Offering may use session cookies that collect each user's name, email address, phone number, or other personally identifiable information for purposes of enhanced user usability and single sign-on configuration. These cookies can be disabled, but disabling them will also eliminate the functionality they enable.

If the configurations deployed for this Software Offering provide you as customer the ability to collect personally identifiable information from end users via cookies and other technologies, you should seek your own legal advice about any laws applicable to such data collection, including any requirements for notice and consent.

For more information about the use of various technologies, including cookies, for these purposes, see IBM's Privacy Policy at ibm.com/privacy and IBM's Online Privacy Statement at ibm.com/privacy/details in the section entitled "Cookies, Web Beacons and Other Technologies," and the "IBM Software Products and Software-as-a-Service Privacy Statement" at ibm.com/software/info/product-privacy.

Policy for unsupported hardware

Various z/OS elements, such as DFSMSdfp, JES2, and MVS, contain code that supports specific hardware servers or devices. In some cases, this device-related element support remains in the product even after the hardware devices pass their announced End of Service date. z/OS may continue to service element code; however, it will not provide service related to unsupported hardware devices. Software problems related to these devices will not be accepted for service, and current service activity will cease if a problem is determined to be associated with out-of-support devices. In such cases, fixes will not be issued.

Minimum supported hardware

The minimum supported hardware for z/OS releases identified in z/OS announcements can subsequently change when service for particular servers or devices is withdrawn. Likewise, the levels of other software products supported on a particular release of z/OS are subject to the service support lifecycle of those

products. Therefore, z/OS and its product publications (for example, panels, samples, messages, and product documentation) can include references to hardware and software that is no longer supported.

- For information about software support lifecycle, see: [IBM Lifecycle Support for z/OS \(www.ibm.com/software/support/systemsz/lifecycle\)](http://www.ibm.com/software/support/systemsz/lifecycle)
- For information about currently-supported IBM hardware, contact your IBM representative.

Trademarks

IBM, the IBM logo, and ibm.com are trademarks or registered trademarks of International Business Machines Corp., 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 \(www.ibm.com/legal/copytrade.shtml\)](http://www.ibm.com/legal/copytrade.shtml).

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

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

Index

A

- abend codes, system [70](#)
- accessibility
 - contact IBM [93](#)
- AI Base Component for IBM z/OS
 - SysAIB component trace [72](#)
- AI Control Interface for IBM z/OS [6](#)
- AI Control Interface task [27](#)
- AI framework, *See* z/OS AI Framework
- AI Framework for IBM z/OS Configuration Workflow [17](#)
- AI model server [5](#)
- AI System Services for IBM z/OS
 - installing [16](#)
 - upgrading [24](#)
- AI-powered WLM batch initiator management [79](#)
- AIB messages [41](#)
- AIB_AISS_LEVEL [75](#)
- AISS, *See* AI System Services for IBM z/OS
- artificial intelligence, batch workload management with [79](#)
- assistive technologies [93](#)

B

- batch initiators, WLM-managed with AI [79](#)
- batch workload management with AI [79](#)

C

- checks
 - AIB_AISS_LEVEL [75](#)
- component trace
 - SysAIB for AI Base Component for IBM z/OS [72](#)
- configuration property file [22](#)
- configuration variables, exporting [19](#), [22](#)
- configuration variables, importing [21](#), [22](#), [24](#)
- configuration workflow, adding a system to z/OS AI Framework [20](#)
- configuration workflow, z/OS AI Framework [17](#)
- configuring (additional system) [20](#)
- configuring (first system) [17](#)
- configuring the z/OS AI Framework [17](#)
- contact
 - z/OS [93](#)
- coupling facility [13](#)

D

- data collection [5](#)
- DFSMS [14](#)

E

- EzNoSQL for z/OS
 - cache structure [13](#)
 - space requirements [14](#)

F

- fix category [39](#)
- FIXCAT [39](#)

H

- hardware requirements [13](#)
- Honor Priority [37](#)

I

- IBM Semeru [14](#)
- IBM Z Common Data Provider [5](#), [7](#)
- IIPHONORPRIORITY [37](#)
- initiators, WLM-managed with AI [79](#)
- installing [15](#)
- IWM messages [89](#)
- IZU messages [58](#)

K

- keyboard
 - navigation [93](#)
 - PF keys [93](#)
 - shortcut keys [93](#)

M

- Machine Learning for IBM z/OS Core Edition [6](#), [7](#)
- messages, AIB [41](#)
- messages, IWM [89](#)
- messages, IZU [58](#)
- MLz Core [6](#)

N

- navigation
 - keyboard [93](#)

O

- operating the z/OS AI Framework [35](#)
- overview [27](#)

P

- packaging, software products [7](#)
- performance considerations [37](#)

- ports, default [15](#)
- prerequisites
 - hardware [13](#)
 - software [14](#)
- product packaging [7](#)
- program directory [15](#)
- property file, configuration
 - exporting variables to [19](#), [22](#)
 - importing variables from [21](#), [22](#), [24](#)
- providers and use cases [77](#)

R

- requirements
 - hardware [13](#)
 - software [14](#)

S

- service
 - when training required [39](#)
- service considerations [39](#)
- shortcut keys [93](#)
- software product packaging [7](#)
- software requirements [14](#)
- Spark runtime instance [9](#)
- START command
 - starting AI Base Component for IBM z/OS) [35](#)
 - starting the ZCDP components) [35](#)
- STOP command
 - stopping AI Base Component for IBM z/OS) [36](#)
 - stopping the ZCDP components) [35](#)
- summary of changes [xiii](#)
- support considerations [39](#)
- SysAIB component trace
 - formatting [74](#)
 - output from [74](#)
 - requesting [72](#)
- system abend codes [70](#)
- system completion codes [70](#)
- system messages [41](#), [89](#)

T

- terminology [3](#)
- trademarks [98](#)

U

- upgrade workflow, AI System Services for IBM z/OS [24](#)
- upgrading (AI System Services for IBM z/OS) [24](#)
- user interface
 - ISPF [93](#)
 - TSO/E [93](#)

V

- VSAM [5](#), [14](#)

W

- workflow [17](#)
- workflow definition file [18](#), [19](#), [21](#), [24](#), [81](#)
- workflow, adding a system to z/OS AI Framework [20](#)
- workflow, AI System Services for IBM z/OS upgrade [24](#)
- workflow, general information [17](#)
- workflow, z/OS AI Framework configuration [17](#)

Z

- z/OS AI Framework
 - adding a system [20](#)
 - components
 - AI model server [5](#)
 - data collection [5](#)
 - providers and use cases [6](#)
 - z/OS AI Base [6](#)
 - z/OS AI Interface [6](#)
 - configuring [17](#)
 - operating [35](#)
- z/OS AI Services, *See* AI System Services for IBM z/OS
- z/OS Management Facility (z/OSMF) [6](#)



Product Number: 5655-ZOS