Redbooks

ibm.com/redbooks

AI on IBM Z: Applications and Examples

Lydia Parziale

Jasmeet Bhatia

Jacky Doll

Khadija Souissi

Markus Wolff

Shin Kelly Yang



IBM Z

Artificial Intelligence

IBM

Redpaper



IBM Redbooks

AI on IBM Z: Applications and Examples

December 2025

REDP-5758-00

Note: Before using this information and the product it supports, read the information in “Notices” on page v.

First Edition (December 2025)

This edition applies to Version 3, Release 1, Modification 0 of IBM z/OS (product number 5650-ZOS)

This document was created or updated on December 12, 2025.

© Copyright International Business Machines Corporation 2025. All rights reserved.
Note to U.S. Government Users Restricted Rights -- Use, duplication or disclosure restricted by GSA ADP Schedule Contract with IBM Corp.

Contents

Notices v
Trademarks vi

Preface vii
Authors vii
Now you can become a published author, too! viii
Comments welcome ix
Stay connected to IBM Redbooks ix

Chapter 1. AI capabilities on IBM Z 1
1.1 Benefits of AI on IBM Z 3
1.2 Advanced AI approaches and applications on IBM Z 5
1.2.1 AI inferencing 7
1.2.2 AI assistants 7
1.2.3 AI agents 8
1.2.4 Product overview and examples 10
1.3 End-to-end support and deployment 17

Chapter 2. Entry points to AI on IBM Z 25
2.1 The AI life cycle methodology 26
2.2 AI on IBM Z products used in the AI lifecycle 27

Chapter 3. Entry Point 1 - IBM turnkey solution examples 29
3.1 AI-enhanced SQL analytics - SQL Data Insights in Db2 for z/OS v13 30
3.1.1 Use case objectives 30
3.1.2 Challenges 30
3.1.3 Solution 30
3.1.4 Additional use cases with SQL Data Insights 31
3.1.5 How it works 31
3.1.6 Best practices for using SQL Data Insights 34
3.2 Intelligent Db2 operations - Db2 AI for z/OS 35
3.2.1 SQL optimization 35
3.2.2 System Assessment and Performance Insights 37
3.2.3 Distributed connection control 39
3.2.4 Best practices for effective use 40
3.3 Operations productivity and skills acceleration - watsonx Assistant for Z 40
3.3.1 Challenge 41
3.3.2 Solution 41
3.3.3 Best practices for effective use 44

Chapter 4. Entry Point 2: Selecting a model using existing assets 45
4.1 When to choose an existing model 46
4.2 AI Solution Templates - AI on IBM Z GitHub examples 46
4.3 IBM Synthetic Data Sets 48
4.4 Large Language Models support with IBM Telum II and Spyre 49
4.5 Getting started 51

Chapter 5. Entry Point 3: Build your own - starting from scratch 53
5.1 Build your own model - starting from scratch 54

© Copyright IBM Corp. 2025.

5.1.1 Data selection for AI model training 55
5.1.2 Model building 56
5.1.3 AI model formatting 58
5.2 Use case example 1: Banking 58
5.3 Use case example 2: Insurance 59
5.4 Other resources to get started with AI on IBM Z 61
5.4.1 Getting started with use case identification - AI on IBM Z Discovery workshop . . 61

Related publications 63
IBM Redbooks 63
Other publications 63
Online resources 63
Help from IBM 63

Notices

This information was developed for products and services offered in the US. This material might be available from IBM in other languages. However, you may be required to own a copy of the product or product version in that language in order to access it.

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, US

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 jurisdictions 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.

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 provide in any way it believes appropriate without incurring any obligation to you.

The performance data and client examples cited are presented for illustrative purposes only. Actual performance results may vary depending on specific configurations and operating conditions.

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.

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 actual people or business enterprises 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.

© Copyright IBM Corp. 2025.

Trademarks

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

The following terms are trademarks or registered trademarks of International Business Machines Corporation, and might also be trademarks or registered trademarks in other countries.

CICS®
Concert®
Db2®
IBM®
IBM Cloud®
IBM Cloud Pak®
IBM Concert®
IBM Research®
IBM Spyre™
IBM Telum®
IBM Watson®
IBM watsonx®
IBM Z®
IBM z16®
Redbooks®
Redbooks (logo) ®
Spyre™
watsonx®
watsonx Assistant®
watsonx Code Assistant®
watsonx.ai®
WebSphere®
z/OS®
z16®
z17™

The following terms are trademarks of other companies:

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

Open Mainframe Project, are trademarks of the Linux Foundation.

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

OpenShift, Red Hat, are trademarks or registered trademarks of Red Hat, Inc. or its subsidiaries in the United States and other countries.

Other company, product, or service names may be trademarks or service marks of others.

Preface

As artificial intelligence (AI) continues to transform industries, organizations are seeking powerful, secure, and scalable platforms to deploy AI workloads. IBM Z®, renowned for its enterprise-grade reliability and performance, is uniquely positioned to meet this demand.

This IBM Redbooks publication introduces readers to the evolving landscape of AI on IBM Z, offering a comprehensive overview of the platform's AI capabilities and strategic value. Whether you're just beginning your AI journey or looking to expand existing initiatives, this book provides clear, actionable entry points tailored to your needs. Explore turnkey solutions from IBM and independent software vendors (ISVs), learn how to integrate AI with your existing assets, or discover how to build custom AI solutions from the ground up.

This IBM Redbooks publication focuses primarily on IBM z/OS®. If you are specifically interested in Linux on IBM Z and LinuxONE, refer to the IBM Redbooks publication, *AI for Linux on IBM Z and LinuxONE applications and examples*, REDP-5756.

With real-world examples, architectural insights, and practical guidance, this book empowers IT leaders, architects, and developers to harness the full potential of AI on IBM Z—driving innovation, efficiency, and competitive advantage in the digital era.

Authors

This paper was produced by a team of specialists from around the world working at IBM Redbooks, Center.

Lydia Parziale is a Project Leader with the IBM Redbooks® team in Poughkeepsie, New York. She brings extensive domestic and international experience in technology management, encompassing software development, project leadership, and strategic planning. Her core expertise lies in business development and database management technologies.

Lydia holds several prestigious certifications, including PMI's Project Management Professional (PMP), IBM Certified IT Specialist, and Open Group Master Certified IT Specialist. She also earned an MBA in Technology Management. With over 30 years at IBM, Lydia has contributed to a wide range of technology initiatives across the organization.

Jasmeet Bhatia is a Product Manager on the AI on IBM Z team, where she plays a pivotal role in enabling enterprises to leverage AI capabilities directly on the IBM Z platform. With over five years of industry experience across artificial intelligence, data science, and product management, she bridges the gap between complex technical innovation and real-world business applications. Her work focuses on helping clients getting started with AI on IBM Z. She collaborates closely with engineering, research, and customer-facing teams to ensure AI solutions are scalable, secure, and optimized.

In addition to her professional work, Jasmeet is also pursuing a doctorate in Business Analytics and Data Science. Her academic research complements her industry work, focusing on advanced analytics and the ethical and responsible deployment of AI in enterprise systems.

© Copyright IBM Corp. 2025.

Jacky Doll has been with IBM for ten years, primarily working on IBM z/OS content design. Jacky helped deliver the first ever IBM Z content solution. She has recently joined the Red Books team as a project leader. Jacky is also pursuing her Ph.D. in cognitive science at RPI.

Khadija Souissi is Principal Solution Architect responsible for AI on IBM Z and IBM LinuxONE solutions. She is based in Germany, holds a Master Degree in Electrical Engineering, is a certified Distinguished Technical Specialist, and has a very rich experience acting as a technical leader in the area of Data and AI on IBM Z, designing and delivering AI and Analytics solutions for IBM Z Clients.

In her previous role as a Principal Product Manager, Khadija has led the delivery of innovative AI solutions such as IBM watsonx Assistant for Z and AI infused capabilities in z/OS. In her current role, she is leading AI on IBM Z and IBM LinuxONE clients' deployments.

Khadija co-leads the Technical Specialist profession in Germany, Austria and Switzerland. She has written extensively on the introduction to AI on Z, the Entry Points, and use case examples.

Markus Wolff is a Technical Specialist for IBM Z solutions in the context of Data and AI on IBM Z since 2019. He holds a Bachelor Degree in Business Administration with focus on Industry 4.0 and Digitization. During his studies, he focused on the digitization of business models and the economical parts of IT Operations Analytics and Machine Learning.

In his current role, he supports clients in Proof of Concepts or client workshops in the context of Data Provisioning, Data Science and Artificial Intelligence in regard to IBM Machine Learning for z/OS, Db2 AI for z/OS, Open Source on IBM Z / LinuxONE and the watsonx portfolio.

Shin Kelly Yang is a Senior Product Manager for AI on IBM Z & LinuxONE in the United States. She has over 10 years of experience in product management and enterprise technology. She holds a degree in B.S. Computer Science and MBA from Clarkson University. Her areas of expertise include artificial intelligence, IBM Z and LinuxONE, and enterprise software strategy. She has written extensively on AI integration in enterprise environments and best open-sourced technology practices.

Thanks to the following people for their contributions to this project:

Calvin Friedrich, Technical Sales Specialist
IBM Germany

Artem Minin
IBM US

Now you can become a published author, too!

Here's an opportunity to spotlight your skills, grow your career, and become a published author—all at the same time! Join an IBM® Redbooks residency project and help write a book in your area of expertise, while honing your experience using leading-edge technologies. Your efforts will help to increase product acceptance and customer satisfaction, as you expand your network of technical contacts and relationships. Residencies run from two to six weeks in length, and you can participate either in person or as a remote resident working from your home base.

Find out more about the residency program, browse the residency index, and apply online at:

ibm.com/redbooks/residencies.html

Comments welcome

Your comments are important to us!

We want our papers to be as helpful as possible. Send us your comments about this paper or other IBM Redbooks publications in one of the following ways:

- Use the online **Contact us** review Redbooks form found at:

  ibm.com/redbooks

- Send your comments in an email to:

  redbooks@us.ibm.com

- Mail your comments to:

  IBM Corporation, IBM Redbooks
  Dept. HYTD Mail Station P099
  2455 South Road
  Poughkeepsie, NY 12601-5400

Stay connected to IBM Redbooks

- Find us on LinkedIn:

  https://www.linkedin.com/groups/2130806

- Explore new Redbooks publications, residencies, and workshops with the IBM Redbooks weekly newsletter:

  https://www.redbooks.ibm.com/subscribe

- Stay current on recent Redbooks publications with RSS Feeds:

  https://www.redbooks.ibm.com/rss.html

1

AI capabilities on IBM Z

In recent years, the evolution of Artificial Intelligence (AI) has accelerated at an unprecedented pace, rapidly transforming concepts from vision to real-world implementation and redefining possibilities for organizations. Emerging trends such as generative AI and agentic AI powered by large language models and intelligent automation place significant demands on IT infrastructure.

As AI models grow in complexity and the number of parameters increases, the need for intensive processing of simultaneous computations has expanded considerably. Additionally, the rising demand for AI workloads is expected to significantly increase data center capacity and energy consumption. Consequently, electricity usage is projected to grow dramatically, with AI workloads consuming a larger share of total data center power.

To support AI at scale responsibly, it is essential to innovate and optimize the underlying infrastructure. These efforts must also address security, governance, and trustworthy AI requirements. Organizations must consider regulatory frameworks such as the EU AI Act, ISO/IEC 42001, and the NIST AI Risk Management Framework to ensure compliance and mitigate risk.

Infrastructure and IT form the foundation for generating and delivering value through AI. If data is considered as the new fuel, infrastructure acts as the engine that drives organizations' AI journeys toward success. Supporting more than 70% of the world's mission-critical business transactions, the IBM Z platform plays a central role. It provides a robust environment with key strengths such as performance, security, resiliency, scalability, and high availability.

IBM Z integrates AI capabilities that enable deployments at scale by infusing real-time insights close to mission-critical data and co-located with applications. This proximity delivers low latency and strong performance, helping organizations make faster decisions and innovate responsibly. These capabilities support trustworthy AI practices by aligning with IBM's principles of transparency, security, and governance-ensuring that AI is deployed in a way that meets both business and regulatory requirements. IBM watsonx.governance further enhances this approach by providing automated tools to manage risk, monitor compliance, and operationalize responsible AI workflows across hybrid environments

© Copyright IBM Corp. 2025.

IBM Z and its operating systems, such as z/OS, serve as the storage and computing infrastructure for business and mission-critical workloads. In the context of artificial intelligence (AI), IBM Z aligns seamlessly with AI generation requirements not only due to its suitable hardware but also because it provides a robust environment for managing and storing high-quality data, an essential factor for AI success.

This chapter presents the benefits of using AI on IBM Z and explains the AI lifecycle methodology. It also provides a high-level overview of the available entry points and associated software for implementing AI on IBM Z.

1.1 Benefits of AI on IBM Z

IBM Z, along with its operating systems such as z/OS, delivers the high-throughput, low-latency infrastructure essential for mission-critical workloads. In the context of AI, IBM Z is uniquely positioned to support modern AI pipelines thanks to its advanced hardware features—most notably the Telum and Telum II processors, which include on-chip AI acceleration, and the IBM Spyre™ PCIe AI Accelerator. These capabilities enable real-time inferencing and model execution directly on the platform. Combined with a secure, high-integrity data environment and support for open-source AI frameworks optimized for the s390x architecture, IBM Z provides a robust foundation for training, deploying, and operationalizing AI models at scale.

The IBM Z platform offers a range of powerful characteristics that make it especially well-suited for AI-infused applications, transactions, and use cases. These include:

- High Availability and Reliability

  IBM Z is engineered for continuous operation, with industry-leading uptime and fault tolerance. This is critical for AI applications embedded in real-time transactional systems, such as fraud detection or customer service automation.

- Scalability and Performance

  With support for massive transaction volumes and parallel processing, IBM Z can scale AI workloads efficiently—whether it's running inferencing models at scale or processing large datasets for training.

- Integrated AI Acceleration

  The Telum processor includes on-chip AI acceleration, enabling real-time inferencing directly on transactional data without offloading to external systems. This reduces latency and improves decision-making speed.

- Secure and Compliant Data Environment

  IBM Z provides end-to-end encryption, secure boot, and advanced access controls. This is essential for AI models that rely on sensitive or regulated data, such as in finance, healthcare, or government sectors.

- Data Proximity and In-Place Analytics

  AI workloads can run close to the data—within the same system—eliminating the need to move large volumes of data across environments. This improves performance and reduces data governance risks.

- Support for Open-Source AI Frameworks

  IBM Z supports popular AI and machine learning frameworks like TensorFlow, PyTorch, and ONNX, optimized for the s390x architecture. This allows data scientists to use familiar tools while leveraging the platform’s unique capabilities.

- Ecosystem Integration

  With offerings like Machine Learning for z/OS (MLz), Cloud Pak for Data, and SQL Data Insights, IBM Z integrates seamlessly into enterprise AI pipelines, supporting everything from model development to deployment and monitoring.

AI-infused enterprise programs are strategic initiatives within an organization that integrate artificial intelligence across business functions, processes, and systems to drive smarter decision-making, automation, and innovation. Rather than treating AI as a standalone tool or project, an AI-infused program embeds AI capabilities into the core of enterprise operations.

A wide range of AI-infused use cases are emerging on the IBM Z platform, each designed to take full advantage of its unique strengths. These use cases leverage IBM Z's core values—such as high availability, security, scalability, and real-time processing—to deliver intelligent, data-driven outcomes directly within mission-critical environments.

AI inference platforms are systems or environments designed to efficiently run trained AI models-specifically for the inference phase of the AI lifecycle. Inference is the process of using a trained model to make predictions or decisions based on new input data. IBM Z is a mainframe platform that supports real-time AI inferencing directly against transactional workloads. Key features include:

- Telum Processor: The industry's first processor with on-chip AI acceleration, enabling low-latency inferencing without offloading data 1.
- In-Transaction AI: AI models can be executed during transaction processing, ideal for use cases like fraud detection, credit scoring, and risk analysis.
- High Security and Availability: Built-in encryption, compliance features, and near-zero downtime make it suitable for regulated industries.

You can drive business value with enterprise AI on IBM Z by leveraging all platform benefits and IBM Z qualities of service, including resiliency, security, performance, scalability, and low-latency inference capabilities. With 100% real-time predictive AI scoring capabilities, scalable performance enables you to drive business growth through effective risk management and cost reduction.

The latest technology advancements enable you to deploy AI projects more efficiently, optimize outcomes, and increase throughput for AI model inference execution. These innovations also allow you to embrace generative AI securely. Some of these advancements and innovations include:

- Telum II processor with integrated AI acceleration: Enables real-time inferencing directly on transactional data without offloading to external systems, reducing latency and improving throughput.
- IBM Spyre™ PCIe AI Accelerator: A purpose-built card designed for high-performance AI workloads, including generative AI. It supports up to 1 TB of memory and advanced compute cores for low-latency, high-throughput AI applications.
- Enhanced AI software stack: Includes Machine Learning for z/OS, SQL Data Insights, and AI Toolkit for IBM Z and LinuxONE, optimized for Telum II and Spyre hardware.
- Support for Large Language Models (LLMs): IBM Z now supports encoder and decoder LLMs for predictive and generative AI use cases, enabling secure deployment of chatbots, assistants, and multi-model AI applications.
- Energy-efficient design: IBM z17 delivers up to 83% less power consumption compared to comparable x86 servers while achieving 7.5× higher AI throughput (up to 450 billion inferences per day at 1 ms response time).

Furthermore, you can scale AI deployments on IBM Z sustainably by optimizing power consumption for AI inferencing. This approach reduces operational costs while helping you benefit from IBM Z's secure AI capabilities to protect sensitive data and models against threats and attacks with trust.

To understand the advancements and features of the IBM Z platform, see the dedicated IBM Redbooks publications in the IBM z17 Redbooks collection: https://www.redbooks.ibm.com/feature/z17.

1.2 Advanced AI approaches and applications on IBM Z

IBM z17 introduces significant enhancements to drive scalable AI performance for transaction processing (see Figure 1-1). IBM z17 systems enable 450 billion AI inferences per day at 1 ms response time, compared to 300 billion with IBM z16®. This 7.5x increase in AI throughput helps you balance workloads during peak demand across eight on-chip AI accelerators, which are available to every core in a central processor complex (CPC) drawer. In IBM z16 systems, each on-chip AI accelerator supports only eight cores. IBM z17™ achieves all of this with 83% less power consumption than comparable x86 servers.

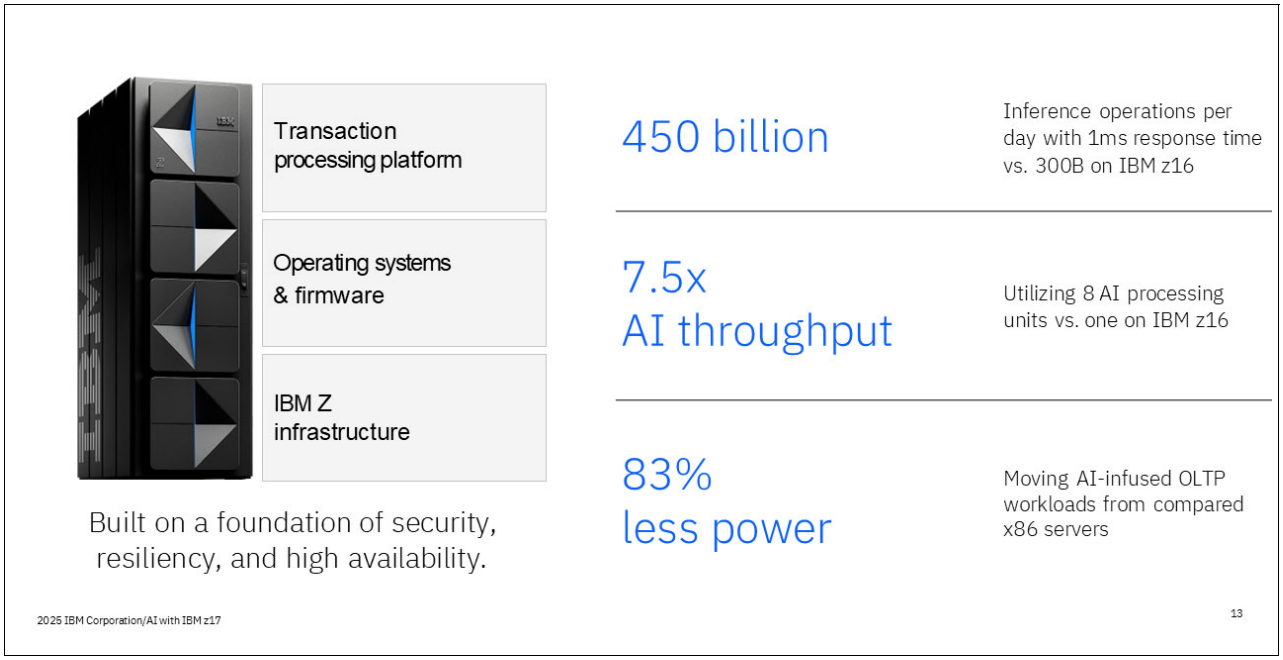


Figure 1-1 AI on IBM z17 advancements

Disclaimer: The performance results presented are extrapolated from IBM internal testing conducted on a z17 LPAR configured with 7 CPs and 256 GB of memory running z/OS v3R1. Testing utilized an IBM proprietary synthetic Anti-Money Laundering detection model leveraging the Integrated Accelerator for AI. The LightGBM model processes graph-structured data combined with pre-generated Know Your Customer (KYC) embeddings using sentence-transformers/all-MiniLM-L6-v2 (https://huggingface.co/sentence-transformers/all-MiniLM-L6-v2) transformer.

IBM introduced on-chip AI acceleration with the Telum processor in the IBM z16 system. The IBM z17 builds on this by incorporating the Telum II processor and IBM intends to deliver the PCIe-based IBM Spyre™ AI Accelerator, which enables IBM z17 to better support generative AI use cases.

Spyre is a purpose-built, enterprise-grade accelerator that offers scalable capabilities for complex AI models and generative AI use cases. It features up to 1 TB of memory and is built to work in tandem across eight cards in a standard I/O drawer to support AI model workloads across the mainframe, whereas each card is designed to consume no more than 75 W. Each chip includes 32 compute cores that support int8 and fp16 data types for low-latency and high-throughput AI applications[1].

IBM z17 introduces significant hardware advancements that enable real-time AI inferencing where your data resides. The Telum II processor integrates an on-chip AI accelerator delivering approximately 24 TOPS INT8 per chip, with intelligent routing across cores for balanced performance. Larger caches—up to 36 MB L2 and virtual L3/L4 totaling several gigabytes—reduce latency and improve data access during inference. A coherently attached Data Processing Unit accelerates I/O, minimizing data movement and enabling single-digit millisecond response times for in-transaction scoring. These capabilities allow you to score every transaction with current features, reducing drift and improving effective model accuracy.

You can also scale AI workloads with the optional Spyre accelerator, designed for emerging models such as LLMs and agent-based architectures. Combined with optimized frameworks like TensorFlow, PyTorch, and SnapML for IBM Z, these enhancements increase throughput, support ensemble models, and maintain SLA compliance. By deploying AI inferencing directly on IBM Z, you improve decision quality, reduce operational complexity, and achieve greater agility for mission-critical applications. For more information on this see: https://www.ibm.com/support/z-content-solutions/spyre-accelerator-z-and-linuxone/

Figure 1-2 illustrates the concepts and application examples for each approach on IBM Z. The following sections provide more details about these concepts and approaches.

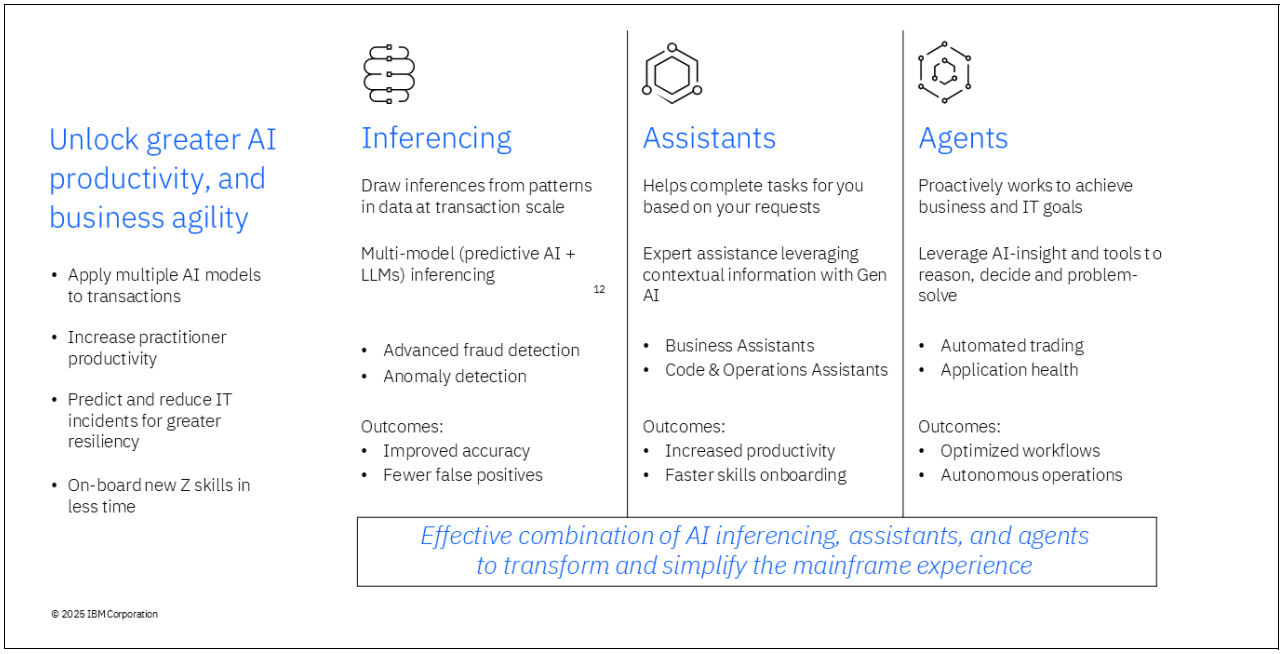


Figure 1-2 Advanced AI approaches and their applications on IBM Z

IBM z17 with the Spyre accelerator enables on-premises generative AI use cases. These include running chatbots, assistants, and agents, along with client-specific applications such as document processing, information search, and data extraction. This approach keeps data secure and in place.[2].

Note: For more information on Telum and Telum II, the on-chip AI acceleration and the Spyre AI Accelerator, see sections 1.1 and 3.3 of the IBM Redbooks publication, *AI for Linux on IBM Z and LinuxONE Applications and Examples*, REDP-5756.

For dedicated information for the Spyre AI Accelerator, please also see Appendix A of the IBM Redbooks publication, *IBM z17 (9175) Technical Guide*, SG24-8579.

[1] https://newsroom.ibm.com/ai-on-z?mhsrc=ibmsearch_a&mhq=spyre
[2] https://www.ibm.com/docs/en/announcements/z17-makes-more-possible

1.2.1 AI inferencing

The AI inferencing segment represents the category of AI that can predict outcomes for you. It includes machine learning and deep learning models and remains essential across several industries. Financial institutions, for example, rely on AI inferencing for use cases such as fraud detection and anti-money laundering (AML). These models are not replaced but are increasingly combined with encoder-based large language models (LLMs) that analyze and extract insights from unstructured content. When combined with predictive AI, they enhance accuracy in scenarios with low confidence levels, such as fraud detection. This combination of predictive AI models and LLMs is known as Multiple Model Architecture. In AI inferencing applications, response times must be extremely short—often within milliseconds—to support real-time transaction processing at scale.

1.2.2 AI assistants

AI assistants represent the category of AI that can chat with users. They provide expert assistance and complete tasks based on your requests by leveraging contextual information with generative AI. Generative AI creates content using models employing the transformer architecture, which are particularly effective at generating text, code, and images.

IBM Z supports the deployment of AI assistants that use generative AI models directly on the platform. These assistants benefit from the platform's built-in capabilities, including hardware acceleration, security, scalability, energy efficiency, and resiliency. In this context, the primary focus is on business assistants, operations assistants, and code assistants.

Business assistants

Business assistants help you gain insights quickly for use cases such as extracting information from documents, analyzing financial records, conducting customer skill screening, and enabling advanced information retrieval. They increase productivity by providing uninterrupted support.

Within generative AI, decoder LLMs typically operate with acceptable response times measured in tokens per second. The primary motivation to deploy assistants on IBM Z is security—ensuring that sensitive documents remain protected within the secure platform while enabling organization-wide access to generative AI capabilities and responsibly scaling workloads.

Operational assistants

IBM Z organizations require access to specialized skills and expertise to fully leverage platform capabilities, enhance performance and security, and optimize business value. The growing demand for experienced IBM Z professionals and the long onboarding process for new users present significant challenges.

IBM watsonx® Assistant for Z is an operational assistant that puts IBM Z expertise and specialized automation skills for repeated and complex tasks at your fingertips. You can use it to leverage platform capabilities, boost productivity and efficiency, and predict and reduce IT incidents for greater resiliency. This offering is especially relevant to system programmers, IT operators, Site Reliability Engineers, and database administrators.

Code assistants

While operational assistants support IBM Z infrastructure teams, code assistants are designed to help application developers with automated and AI-assisted capabilities that accelerate the application lifecycle.

IBM watsonx Code Assistant for Z is a code assistant that uses generative AI and automation to help you explain applications and selectively transform them into well-architected, high-quality code. It supports various phases of the application lifecycle, including discovery, analysis, refactoring, code generation, optimization, and transformation to newer languages. This reduces modernization risks, improves developer productivity, and shortens time to value.

It includes fine-tuned generative AI for mainframe code that helps you modernize faster using an incremental approach. This reduces cost and risk while supporting full mixed-language interoperability.

1.2.3 AI agents

Current AI trends focus on generative AI combined with autonomous and intelligent automation to perform tasks on your behalf. AI agents are central to the trend of combining generative AI with autonomous and intelligent automation.

AI agents defined

AI agents represent an implementation pattern for applications that use LLMs connected to tools and environments. These agents autonomously make decisions and perform tasks with minimal human intervention. They proactively work to achieve business and IT goals by leveraging AI insights and tools to reason, decide, and solve problems.

What sets AI agents apart from generative AI—beyond their ability to make decisions, solve problems, and interact with external environments—is their capacity to perform tasks on behalf of a prime or user and share details of their decision-making process. In other words, AI agents can “think” and take actions using the tools available to them.

Another key difference is that you can assign AI agents high-level goals, such as applying automated trading. In contrast, most LLM usage involves simple, task-oriented prompts and responses. The anticipated outcome of implementing AI agents is optimized workflows and autonomous operations.

Single vs. multiple agents

For complex workflow execution, AI agents use tools and collaborate with multiple agents to perform sophisticated tasks. Each agent handles a specific subtask required to achieve the overall goal, based on a defined plan and workflow logic in response to a question or prompt.

This type of collaboration is referred to as a multiple-agent system. Within such systems, agents can perform steps in parallel or in sequence. They share execution status, outputs, and outcomes with each other, or feed insights to an orchestrator agent that compiles, interprets, and coordinates the information gathered from the different agents.

Figure 1-3 illustrates the difference between a single agent implementation vs. multiple agent implementation. While a single AI agent implementation allows you to perform specific tasks and achieve specific goals based on predefined rules, different agents can work together in a multiple agent approach to execute complexer tasks.

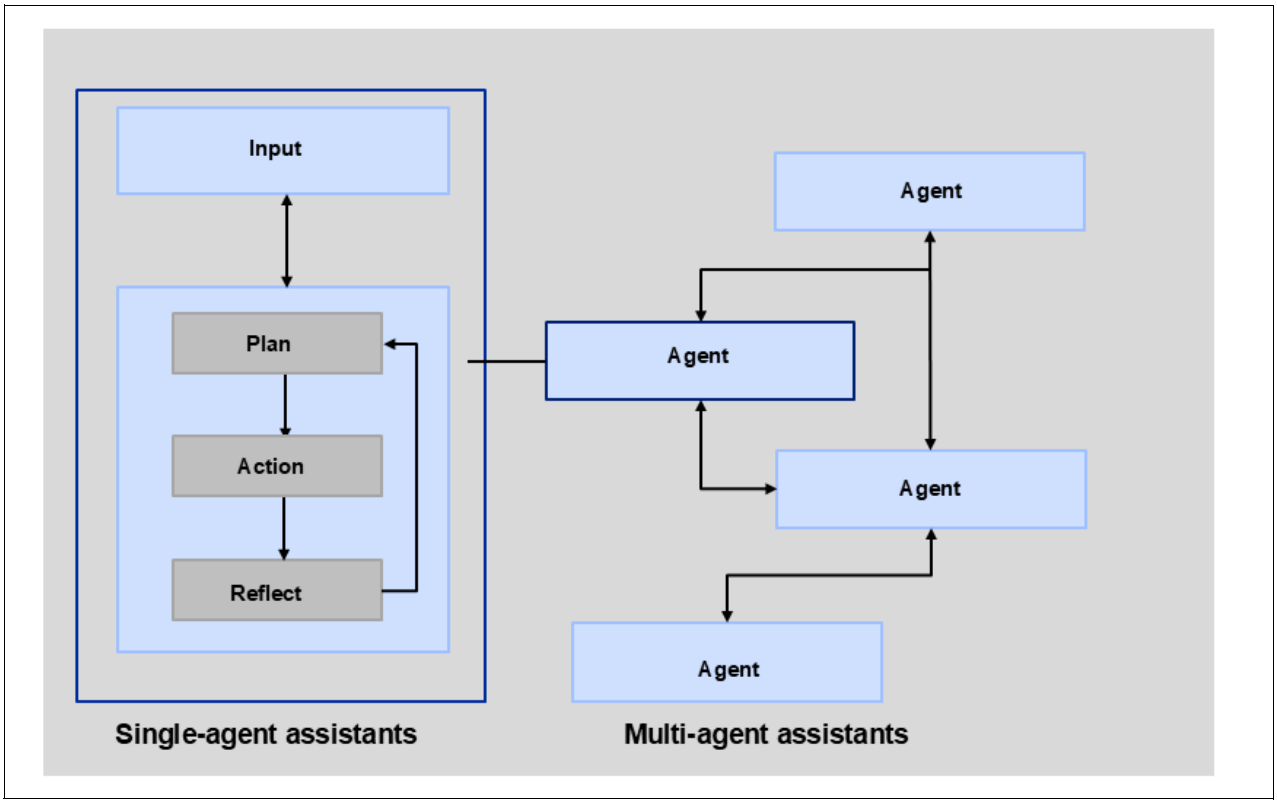


Figure 1-3 Single-agent assistants vs. multi-agent assistants

Agentic AI

Agentic AI is the overarching concept for AI systems that can act independently and achieve goals without constant human oversight. You can also use the term agentic AI systems to describe frameworks that integrate, manage, and orchestrate multiple AI agents. These systems enable agents to collaborate seamlessly within a workflow, supporting intelligent automation of complex processes with limited supervision.

Agentic AI systems represent a shift toward more autonomous solutions that handle complex scenarios requiring continuous learning and adaptation. While AI agents excel at performing narrow, well-defined functions, agentic AI systems proactively respond to changes in their environment, allowing them to navigate uncertainty effectively. This capability is essential for applications such as real-time financial trading.

In contrast, AI agents are typically reactive. They respond to specific inputs but do not anticipate changes or adjust their behavior dynamically.

AI agents and IBM Z

Applying AI agents opens new opportunities for business applications to drive autonomous operations in areas such as customer service, insurance claim processing, and financial management, including automated trading. You can also deploy AI agents to address common customer service issues without human intervention.

Deploying AI agents on IBM Z for these use cases shows strong potential to leverage the key benefits of AI applications co-located with the many connected tools that run on the platform. This integration supports intelligent automation and enhances operational efficiency across complex workflows.

When it comes to operations, the latest capabilities of IBM watsonx Assistant® for Z leverage the concept of AI agents to autonomously perform tasks on behalf of a user or system. These agents execute multi-step processes with minimal human intervention, including autonomous actions and self-corrections.

In this context, AI agents can respond to alerts and events, proactively perform health checks, and ensure that system performance continues to meet expected goals. You can also apply AI agents to platform operations, enabling them to perform tasks such as adding users, managing networks, configuring storage groups, and adjusting priority classes.

1.2.4 Product overview and examples

This section provides a high-level overview of available AI application areas and offerings on IBM Z.

Figure 1-4 highlights two of the focus areas of AI on IBM Z:

- Business insights

  AI inference use cases for IBM Z include

  - Fraud detection
  - Customer behavior prediction
  - Supply chain optimization

  These use cases leverage IBM Z's ability to deliver real-time insights directly within mission-critical environments.

- Intelligent infrastructure

  AI capabilities on IBM Z also support intelligent infrastructure optimization, including:

  - Workload placement
  - Database query planning
  - Anomaly detection for security

  These entry points are briefly explained in the following sections, along with the associated IBM software offerings that support each use case.

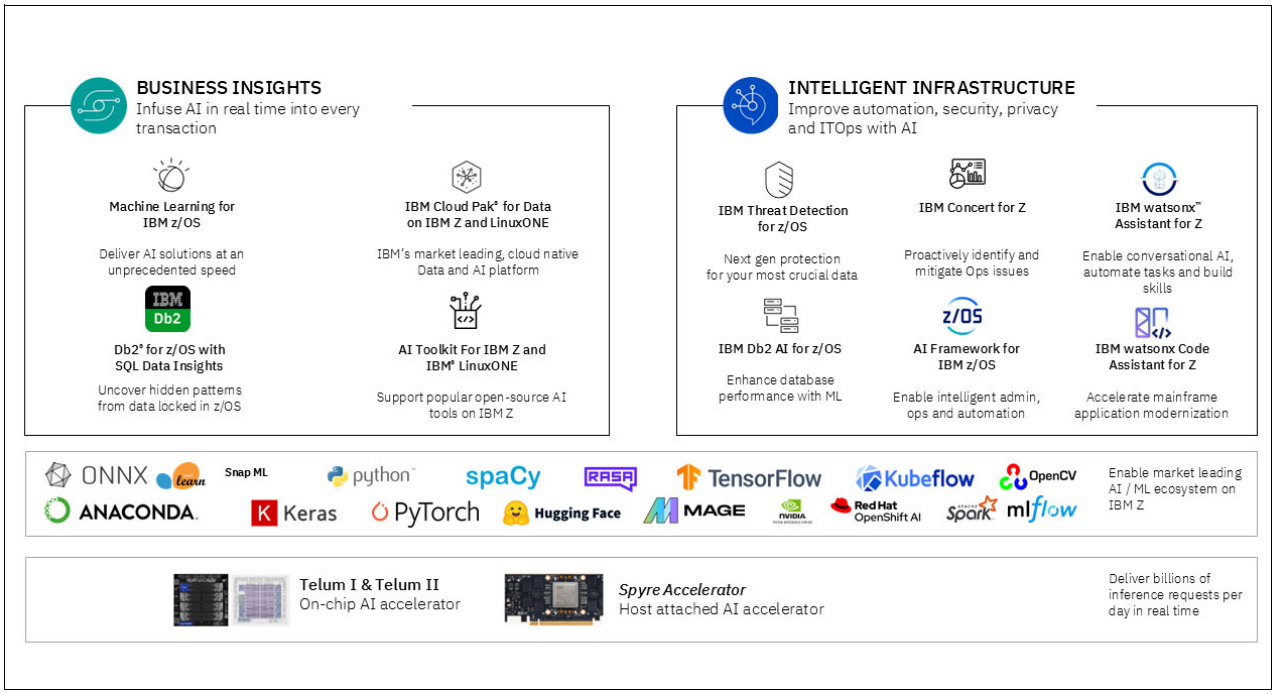


Figure 1-4 Focus areas within the AI on IBM Z ecosystem stack

AI ecosystem and open-source

One of the foundations that supports these two focus areas—and their associated software and use cases—is one of the market-leading AI ecosystems, which includes a variety of open-source technologies. IBM and open-source communities adapt AI frameworks and Python packages to the IBM Z architecture (s390x) and enable them to take advantage of hardware acceleration. Figure 1-5 shows a subset of available AI frameworks and Python packages at various stages of exploiting IBM Z hardware capabilities at the time of publishing this IBM Redbooks publication. This list continues to evolve and update regularly.

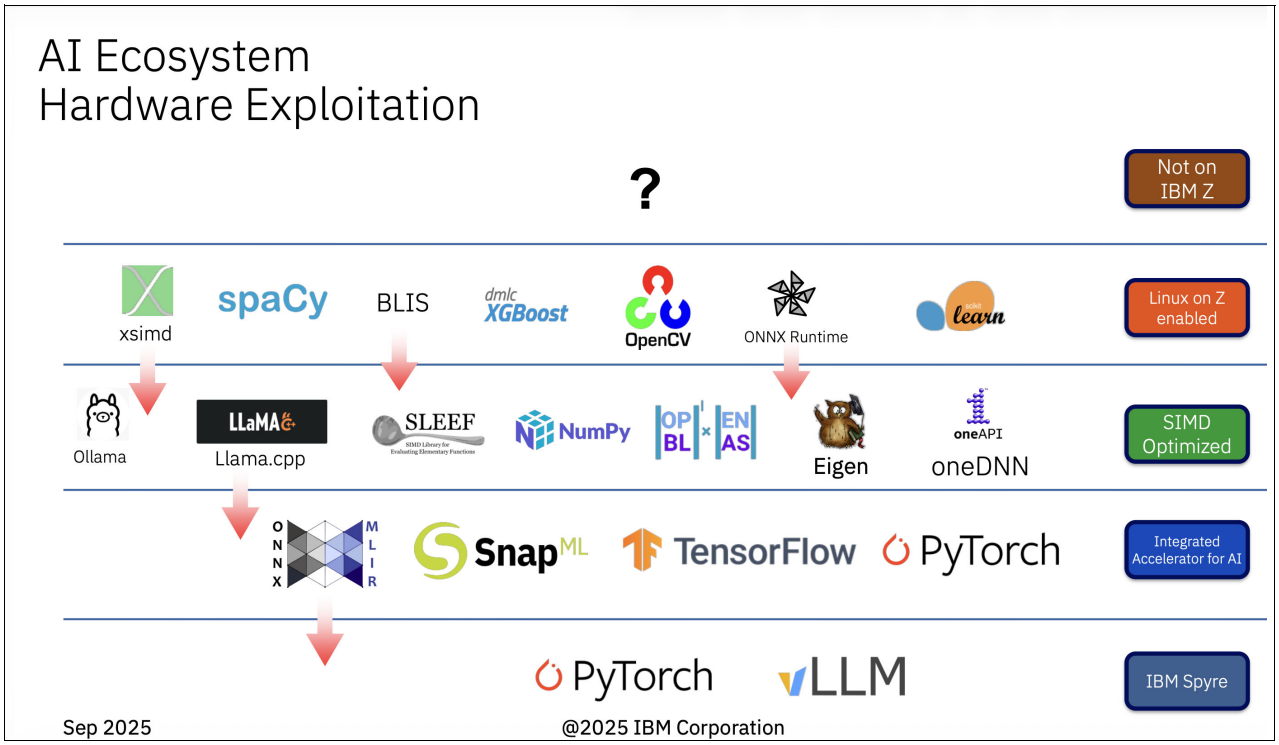


Figure 1-5 Subset of available AI frameworks and Python libraries on IBM Z

In addition to well-known sources for AI open-source software, such as PyPI and Docker Hub, IBM provides its own repositories to help you avoid the challenges listed in Figure 1-6. The container images and packages available through IBM's repositories are IBM-certified and undergo strict security checks.

Note: For more information on IBM's AI open-source repositories and the security mechanism, see the IBM Redbooks solution guide, *Securely Leverage Open-Source Software with Python AI Toolkit for IBM z/OS*, REDP-5709.

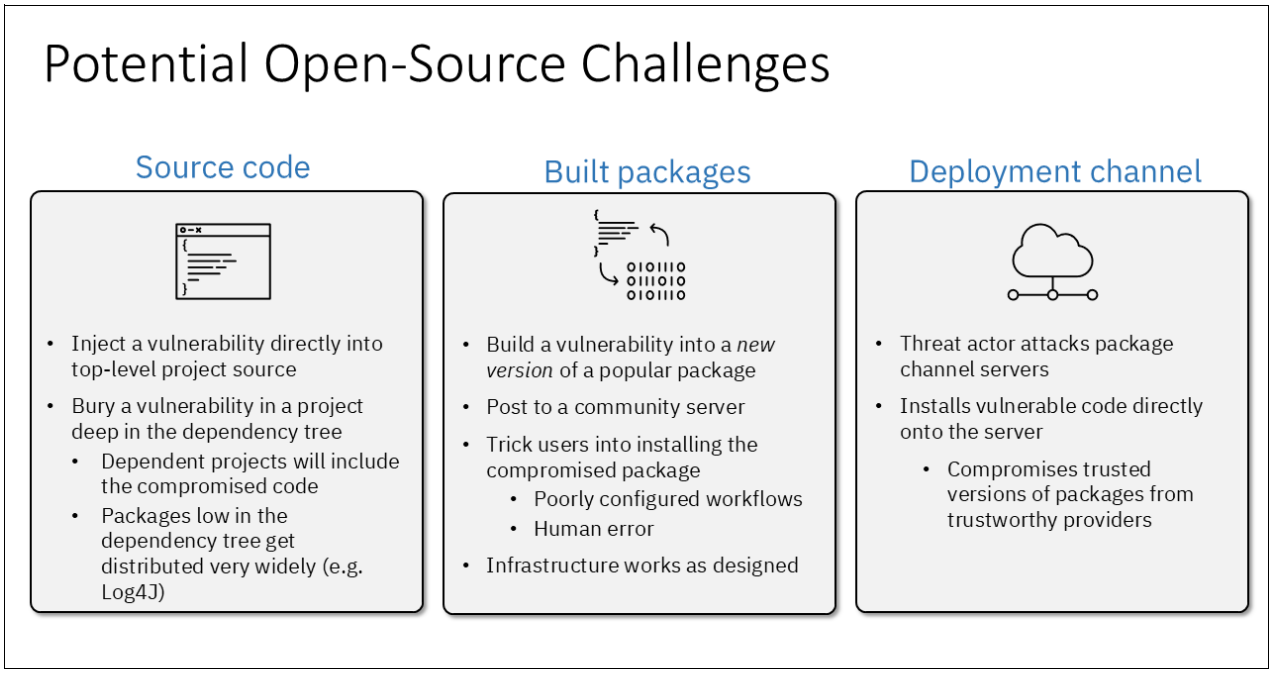


Figure 1-6 Potential open-source challenges

IBM provides certified open-source container images and Python packages—often at no charge—through the channels shown in Figure 1-7. You can purchase additional IBM support if needed.

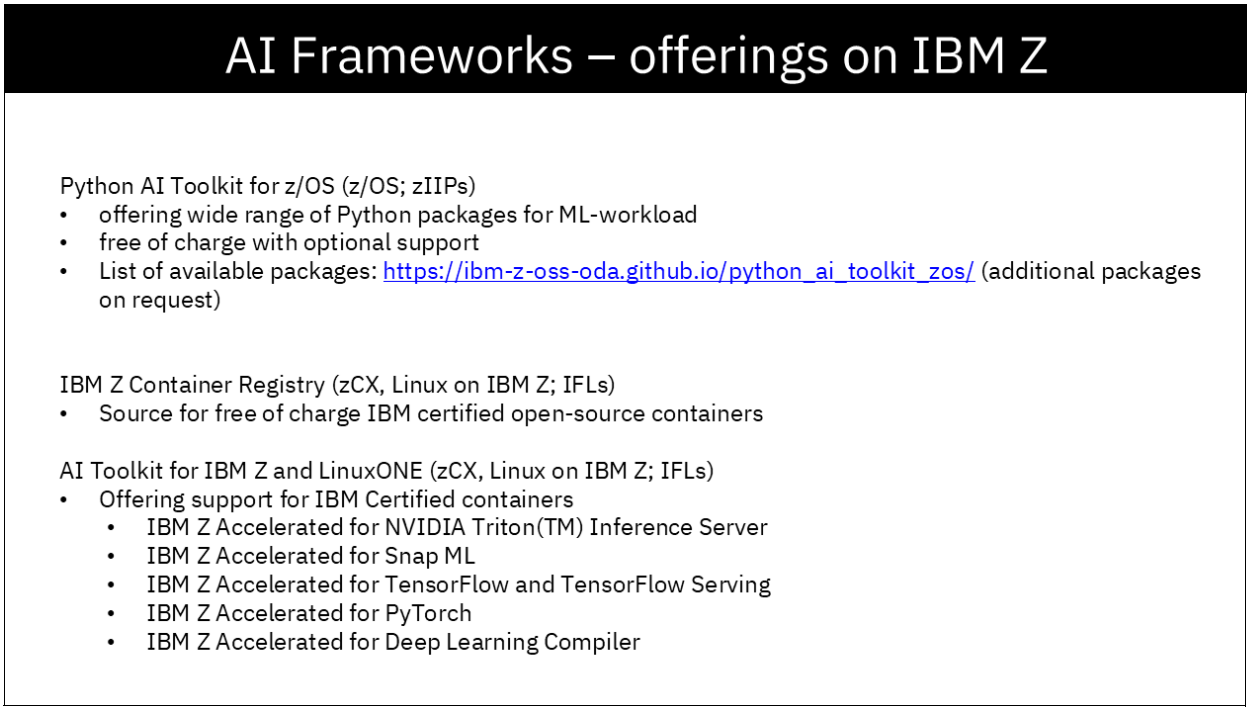


Figure 1-7 IBM's open-source AI repositories

Note: For more information about IBM's open-source AI repositories, see the following resources:

- Python AI Toolkit for z/OS
- IBM Z Container Registry
- AI Toolkit for IBM Z and LinuxONE

Hardware acceleration

Section 1.2, “Advanced AI approaches and applications on IBM Z” on page 5 explains hardware acceleration, including AI Acceleration in Telum II and Telum, and the Spyre PCIe AI Accelerator.

Business insights

IBM Z provides a comprehensive portfolio of AI and machine learning capabilities that help you unlock business insights, improve operational efficiency, and accelerate decision-making. Key solutions include enterprise-grade machine learning with MLz, AI-enhanced SQL queries through SQL Data Insights, containerized AI services with Cloud Pak for Data, and secure open-source integration using the Python AI Toolkit. The AI Toolkit for IBM Z and LinuxONE offers optimized container images and supports advanced inferencing with Telum II and the IBM Spyre Accelerator. These offerings enable you to deploy and scale AI models close to your data and transactions, helping you address use cases such as fraud detection, anti-money laundering, and claims processing with greater accuracy and speed.

- Machine Learning for IBM z/OS Enterprise Edition

  Machine Learning for IBM z/OS (MLz) is an enterprise solution for deploying and inferring machine learning and deep learning models near z/OS assets and transactions. You can import models in common exchange formats or optionally develop them with MLz. This flexibility allows data scientists to continue using their preferred development environments through model export. You can integrate and execute AI models during transaction runtime by using IBM CICS® LINK or IBM WebSphere® Local Adapter (WOLA), integrating SnapML when appropriate, and leveraging the AI Accelerator.

  Note: IBM developed SnapML to accelerate machine learning models and take advantage of the AI Accelerator in Telum I and II.

  Deep learning models in the standardized ONNX format benefit from the AI Accelerator on Telum I and II without requiring SnapML. You must use the included Deep Learning Compiler (zdlc) to run ONNX models in MLz.

  Note: AI Solution Templates are templates for various use cases and can be used with MLz. For more information, see: https://ambitus.github.io/aionz-solution-templates/.

  As part of continuous deliveries, key software products like IBM watsonx Assistant for Z, AI Toolkit for IBM Z and IBM LinuxONE, and Machine Learning for IBM z/OS will use Spyre accelerators for full on-prem deployment.. MLz will leverage the acceleration capabilities of Telum II and IBM Spyre Accelerator, opening new opportunities to deploy AI models for advanced inferencing at scale. With z17, MLz will enable clients to leverage encoder LLMs (like BERT) that excel at understanding natural language and processing unstructured text, focusing on extracting key information for improved scoring results (e.g., increased accuracy, reduced false positives, etc.) and address a wider range of advanced AI use cases like fraud detection, anti-money laundering (AML), insurance claims, etc. to achieve business insights at scale.[3]

- Cloud Pak for Data on IBM Z and LinuxONE

  Cloud Pak for Data on IBM Z and LinuxONE (CP4D) is a solution offering containerized services around Data and AI, like setting up a Data Fabric, creating AI models or deploying them.

  Note: For a detailed description of CP4D and its included features, see section 2.2 in the IBM Redbooks publication *AI for Linux on IBM Z and LinuxONE Applications and Examples*, REDP-5756 .

- SQL Data Insights

  SQL Data Insights (SQL DI) is a no-charge component of the latest Db2 for z/OS version 13. SQL DI enables you to run straightforward, AI-enhanced semantic queries using standard SQL. You can call AI functions directly from existing applications that support SQL.SQL DI offers various semantic queries, such as similarity search, which helps you find entities similar to a selected one. Because you don't need to move data, you can securely train and run models directly in the database. You also do not need data science skills to use it..

  SQL DI is typically a good entry point in AI on IBM Z.

  Note: For more information on SQL DI, see Chapter 3, "Entry Point 1 - IBM turnkey solution examples" on page 29, chapter 6 in *IBM Db2 13 for z/OS and More*, SG24-8527" and Running AI queries with SQL Data Insights.

- Python AI Toolkit

  IBM provides its own repository of Python packages, especially for machine learning applications. IBM certifies these no-charge Python packages, which go through rigorous security checks. PyPI is a popular source of Python packages among data scientists. With the Python AI Toolkit, IBM addresses PyPI security concerns and ensures that the packages are optimized for the IBM Z architecture.

  Additional, optional IBM support is available.

  Note: For more information on the Python AI Toolkit, see Securely Leverage Open-Source Software with Python AI Toolkit for IBM z/OS .

- AI Toolkit for IBM Z and LinuxONE

  IBM offers no-charge, certified container images through the IBM Container Registry, including images for developing and running AI applications. These container images are IBM certified and go through rigorous security checks. They are also optimized for hardware acceleration on the IBM Z platform, including SIMD and the AI Accelerator on Telum and Telum II.

  The AI Toolkit provides IBM-supported containers from the IBM Container Registry, specifically for AI workloads. You can obtain frameworks such as TensorFlow and PyTorch this way. For multi-model AI applications—combining machine learning or deep learning models with LLMs—IBM already provides open-source components adapted for IBM Z.

  Software support for running encoder and decoder LLMs on the Spyre AI Accelerator will be available on IBM Z and LinuxONE. In addition, software to run encoder LLMs on the integrated AI Accelerator in Telum II will also be provided for IBM Z and LinuxONE.

[3] https://www.ibm.com/docs/en/announcements/z17-makes-more-possible

MLz and the AI Toolkit for IBM Z and LinuxONE both include the latest version of the IBM Z Deep Learning Compiler which allows the exploitation of Telum II capabilities. In addition, enhancements have also been made to IBM Z Accelerated for PyTorch, IBM Z Accelerated for TensorFlow, IBM Z Accelerated Serving for TensorFlow, and IBM Z Accelerated for NVIDIA Triton Inference Server, optimized to take full advantage of Telum II.

OpenShift AI on IBM Z and LinuxONE is one of the initial offerings that use IBM Spyre technology on IBM Z to deliver AI model serving and inference capabilities. This software stack enables you to run AI workloads securely and efficiently where your critical data resides.

OpenShift AI combines OpenShift container orchestration with enterprise-grade AI/MLOps tools. You get a cloud-like experience for developers and data scientists. It uses IBM Telum II processors and Spyre accelerators to support low-latency inference and scalable generative AI, and it integrates trusted AI frameworks like Trustworthy AI for governance. The solution helps you accelerate insights, reduce infrastructure costs, and maintain security without compromising innovation.

OpenShift AI provides the technical foundation for upcoming IBM solutions, such as watsonx.ai.

The Red Hat AI Inference Server, now supported on IBM Z and LinuxONE, delivers fast, cost-effective AI inference. On IBM Z and LinuxONE, you can use it as part of Red Hat OpenShift AI or as a standalone product. It integrates seamlessly with Red Hat OpenShift AI to provide scalable, portable, and optimized inference capabilities for enterprise AI workloads.

As a standalone product, it brings optimized LLM inference capabilities with true portability across your hybrid cloud environments. AI Inference Server provides you with a hardened, supported distribution of vLLM. The solution offers flexibility to run models on-premises, including IBM Z and LinuxONE, in the cloud, or at the edge, and includes a repository of validated open-source models as part of the Red Hat AI platform.

When you deploy it on IBM Z and LinuxONE, IBM Spyre is supported. For details, see Supported product and hardware configurations.

Intellingent infrastructure

IBM offers a range of AI-powered software solutions to support intelligent infrastructure. Among those, include:

- IBM Threat Detection for z/OS

  IBM Threat Detection for z/OS uses AI to identify anomalies in data access, enhancing security and compliance with emerging regulations like DORA for IBM Z.

- IBM Concert® for Z

  IBM Concert for Z is an AI-powered IT operations management hub that transforms how organizations manage mainframe resilience by aggregating siloed data and insights to act faster and simplify operations.

- IBM watsonx Assistant for Z

  IBM watsonx Assistant for Z (WA4Z) offers a conversational AI experience that enables different personas and roles to ask IBM Z–specific questions. You can upload client-specific data, such as naming conventions, to enrich the assistant's knowledge base. WA4Z also provides automation capabilities that allow you to execute tasks directly from the conversation and create custom skills to streamline operations.

- IBM Db2® AI for z/OS

Db2 AI for z/OS (Db2ZAI) utilizes machine learning to analyze patterns in your unique operating environment, thereby enhancing the performance of Db2 for z/OS applications and systems. This approach focuses on the following key areas:

- SQL optimization: Utilizes machine learning to assist the Db2 Optimizer in selecting the most efficient access path.
- System assessment: Employs artificial intelligence to identify normal behavior, perform exception-based analysis, and provide performance tuning recommendations at the system level.
- Performance Insights: Employs machine learning for interactive Db2 performance analysis at the application level.
- Distributed connection control: Utilizes artificial intelligence to learn the distributed connection behavior, enabling the control of connection and thread handling.

Note: For more information on Db2ZAI, see Chapter 3, “Entry Point 1 - IBM turnkey solution examples” on page 29

- AI Framework for IBM z/OS

  IBM introduced the underlying framework for AI functions in z/OS starting with z/OS 3.1.

  The AI Framework for IBM z/OS (z/OS AI Framework) brings intelligence to z/OS by optimizing IT processes, streamlining system management, reducing skill requirements, and enhancing performance. It includes components for data collection, model training, deployment, monitoring, and AI inferencing and scoring. The framework also enables seamless integration with other z/OS components by providing traditional z/OS programming interfaces—such as assembler services—to access REST APIs, simplifying AI deployment..

- IBM watsonx Code Assistant® for Z

  The IBM watsonx Code Assistant for Z (WCA4Z) is purpose-built to accelerate mainframe development and application modernization. At the time of publishing, these major use cases are available:

  - Modernize, understand and explain COBOL
  - Transform COBOL to Java
  - Modernize, understand and explain PL/I
  - Understand and explain JCL

1.3 End-to-end support and deployment

IBM Z is used to support business-critical transactions, offering features such as high availability and encryption designed to meet emerging security standards. It plays a central role in many enterprise IT environments. With the IBM z17 hardware generation, machine learning capabilities have been integrated to enable real-time analysis of transactional data. This allows processing to occur without moving data off-platform, which helps reduce latency and potential exposure to data breaches.

Note: For an introduction to the IBM Z platform, including the z/OS operating system and its associated components, refer to the IBM z/OS Basic Skills documentation available at: https://www.ibm.com/docs/en/zos-basic-skills

To illustrate the value of AI on IBM Z—particularly in transactional environments—Figure 1-8 introduces a sample application of a payment transaction flow, where:

1. Inbound Transaction – A transaction is initiated, for example, from an ATM.
2. Application Processing – The request is processed by business logic implemented in a COBOL application running within the CICS transaction monitor.
3. Data Management – The application interacts with Db2 for z/OS to perform updates on business-critical data stored in the relational database.

This end-to-end flow demonstrates how AI can be integrated seamlessly into transactional systems on IBM Z, enabling real-time insights without disrupting core processing.

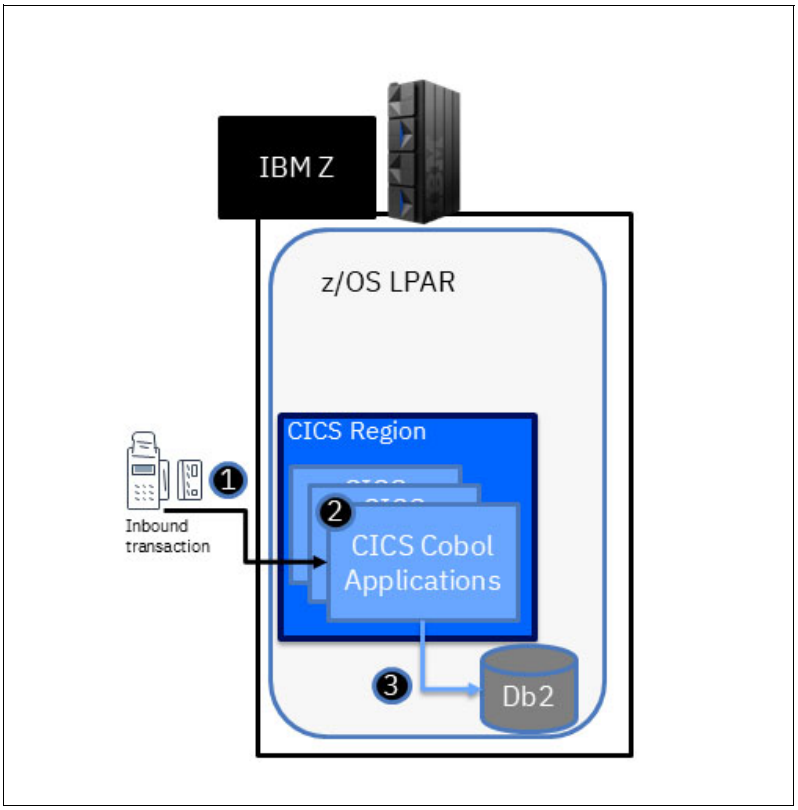


Figure 1-8 Sample end-to-end flow of a payment transaction running on IBM z/OS

In the banking sector, implementing a fraud detection model has become increasingly important due to the rising volume and sophistication of fraudulent activities.

Note: For more information on fraud detection and the reason to care about, see https://www.ibm.com/think/topics/fraud-detection.

Traditionally, fraud detection relied on rule-based systems, which are often implemented as decision trees. As shown in Figure 1-9, these systems are relatively simple to develop and deploy but struggle to adapt to evolving fraud patterns, as they depend on predefined, static rules and relationships. In addition to the payment transaction flow, we see an addition piece of the flow in this figure, (4) rule-based fraud detection, where the transaction is evaluated by a fraud detection service that uses predefined rules.

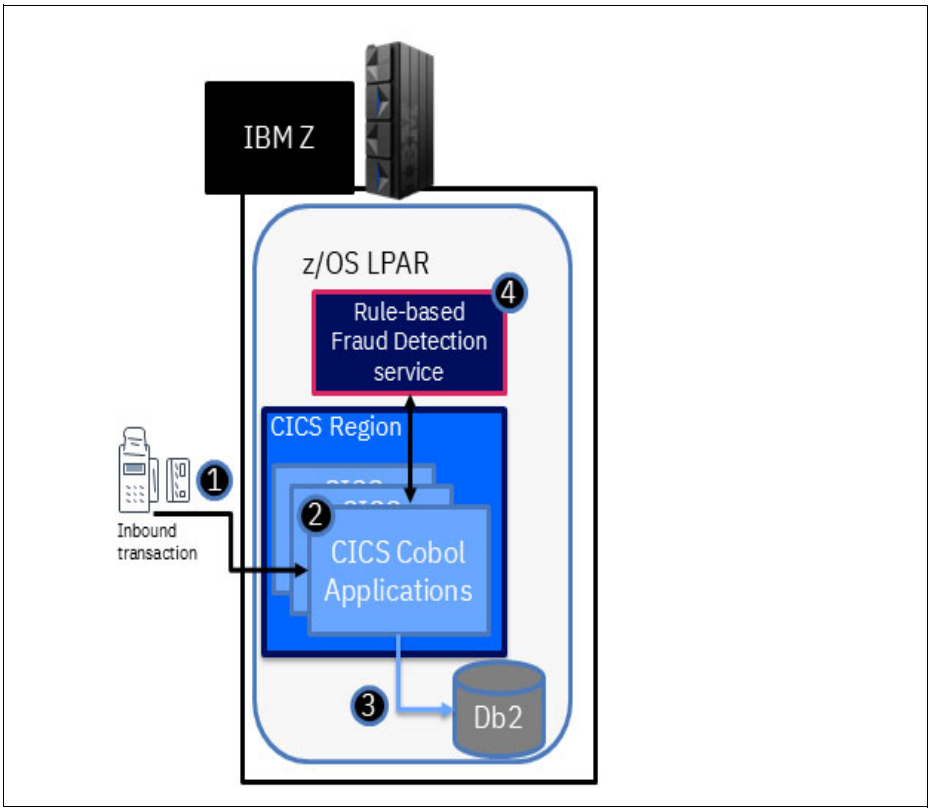


Figure 1-9 Sample flow of a payment transaction running on IBM z/OS expanded by a rule-based fraud detection service

AI-infused fraud detection systems build upon traditional rule-based approaches by leveraging advanced pattern recognition capabilities. These systems can identify complex and previously unseen relationships within large datasets and are designed to scale effectively. However, they are typically more complex to implement.

Note: For more information on the difference between traditional and AI-powered fraud detection, see https://www.ibm.com/think/topics/ai-fraud-detection-in-banking.

While AI services are often deployed on distributed platforms or in cloud environments, IBM Z also supports AI workloads natively. However, many transactions processed on IBM Z are time-sensitive and involve sensitive data, making external API calls—for example, to a cloud-based fraud detection service—problematic. These calls can introduce latency that may violate service-level agreements (SLAs) and delay transaction completion. Additionally, relying on off-platform AI services can result in missed fraud detection opportunities due to timing constraints. In highly regulated industries, moving sensitive data off-platform also raises compliance and security concerns. Figure 1-10 illustrates a sample transaction flow that highlights these challenges.

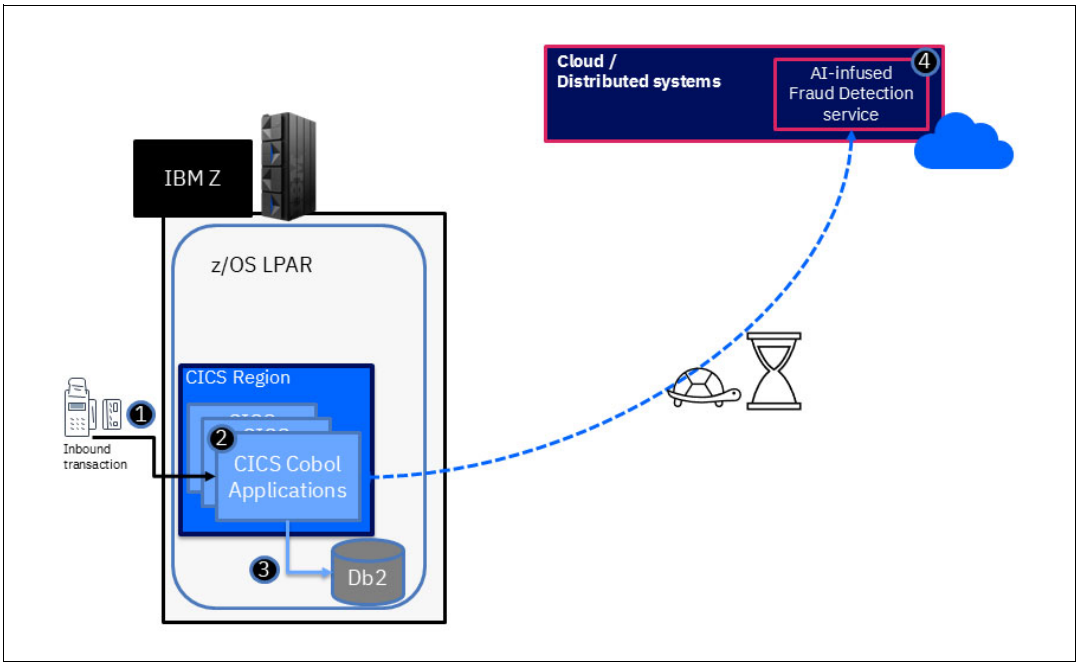


Figure 1-10 AI-infused fraud detection workflow using IBM Z and cloud services showing latency introduced by off-platform processing, highlighting slower response compared to on-platform inference

To benefit from AI-infused fraud detection while avoiding the latency and data movement associated with off-platform processing, it is recommended to deploy the AI model as close as possible to the transaction and data[4]. Running the model directly on IBM Z—integrated with z/OS assets—minimizes response time and eliminates the need to transfer sensitive data, helping maintain strict service-level agreements (SLAs)[5]. Figure 1-11 illustrates a sample flow demonstrating this on-platform deployment approach.

[4] This approach is also called Data Gravity, https://community.ibm.com/community/user/blogs/wagner-cendra/2024/09/10/unlocking-ai-use-cases-with-ibm-z-and-the-new-telu

[5] With IBM z17, processing up to 450 billion inference operations per day with 1 ms response time for real-time use cases is possible, https://www.ibm.com/z/artificial-intelligence https://www.ibm.com/z/artificial-intelligence#footnotes1

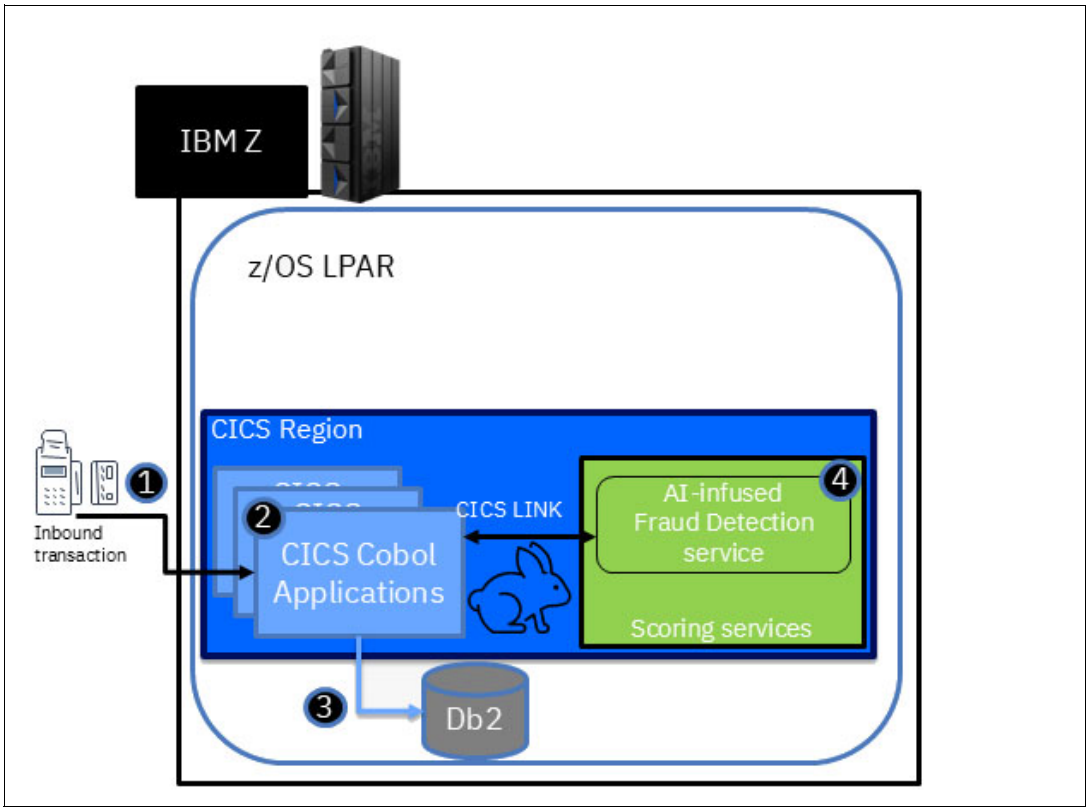


Figure 1-11 Sample flow of a payment transaction running on IBM z/OS expanded by an AI-infused fraud detection service running on IBM z/OS, close to the transactions and data

AI on IBM Z benefits significantly from its comprehensive support for end-to-end AI development and deployment. Its capabilities, highlighted in Figure 1-12, are designed not only to enhance existing transactions and enterprise assets by infusing them with AI, but also to support a broad range of open-source tools and frameworks that leverage the strengths of the IBM Z infrastructure.

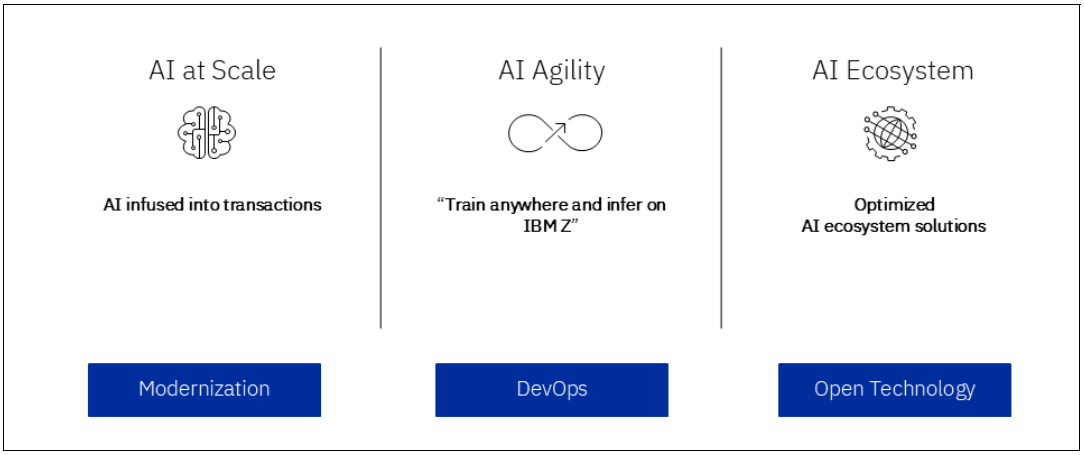


Figure 1-12 AI on IBM Z capabilities

In addition to these foundational elements, a key element is the "Train anywhere, infer on IBM Z" approach, which enables flexible model development across platforms while taking advantage of IBM Z's secure and high-performance inference capabilities.

Typically, specialized departments having employees with skills around data preparation and AI models, like Data Scientist, are the ones creating and training AI models. Often these personas have existing frameworks and environments, which they prefer for their work. As the AI on IBM Z technologies supported a wide variety of frameworks and common exchange format, personas, like Data Scientist can continue using their frameworks and environments and bring developed AI models to IBM Z to deploy them. This deployment is often called inferencing.

Figure 1-13 illustrates the basic flow of the "train anywhere, inference on IBM Z" approach within the AI ecosystem for machine learning.

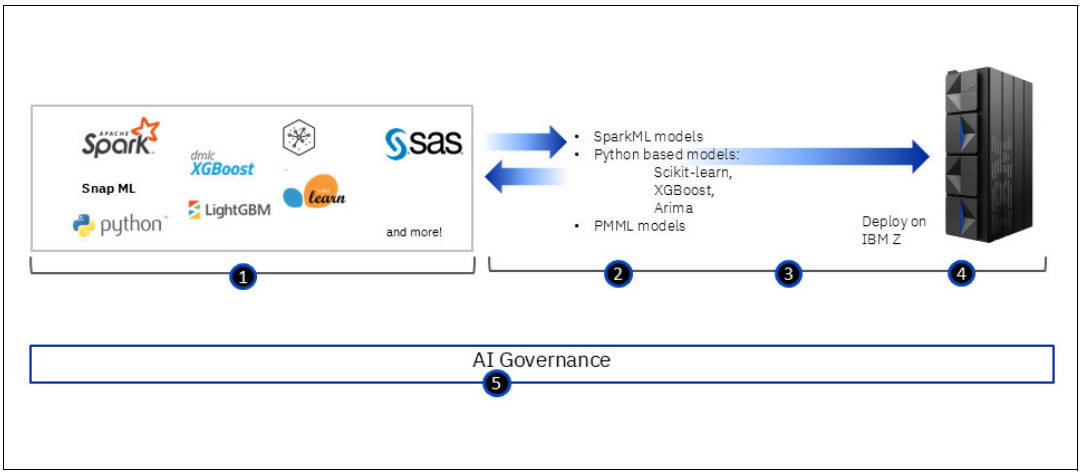


Figure 1-13 Basic flow of the approach "train anywhere and inference on IBM Z

1. Train Anywhere: You can develop models using a wide range of frameworks and libraries, such as Apache Spark, Python, XGBoost, LightGBM, SAS, and SnapML. These tools typically run in diverse environments—on-premises, in the cloud, or on distributed systems—showing that training is not limited to IBM Z.
2. Model Portability: You can export models created with tools such as SparkML, Scikit-learn, and PMML in standard formats for deployment. This capability enables you to train models externally and prepare them for execution on IBM Z..
3. Transition to IBM Z Execution: This step focuses on moving from model development—using tools like SparkML, Scikit-learn, or XGBoost—to preparing those models for execution within the IBM Z environment.
4. Deploy on IBM Z: After preparation, you can deploy models on IBM Z to run inference close to your data and transactions, leveraging the platform's performance and security.
5. AI Governance: The governance layer spans the entire lifecycle—from training to inference—ensuring compliance, monitoring, and management. This is critical when models are trained outside IBM Z and deployed internally.

Figure 1-14 illustrates the end-to-end process of building, training, and deploying AI models using IBM infrastructure, specifically IBM z17 and LinuxONE 5.

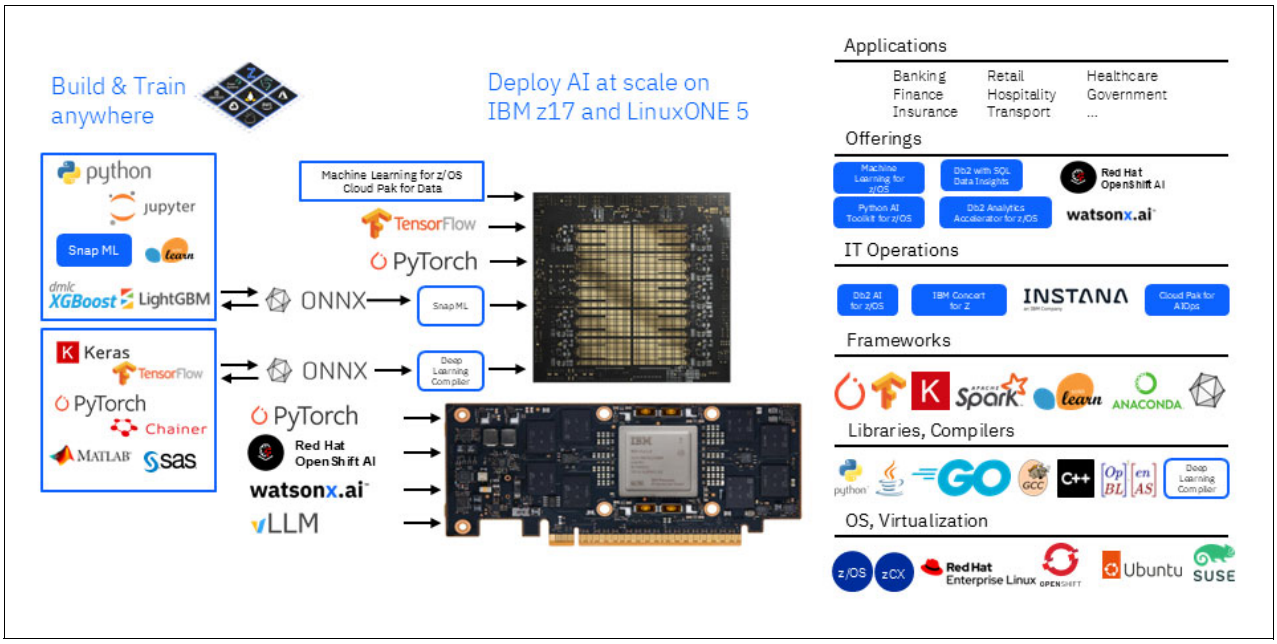


Figure 1-14 General overview of AI on IBM Z

- Left Section – Build & Train Anywhere

  This section highlights the flexibility of using a wide range of tools and frameworks for AI model development, including:

  - Languages/Frameworks: Python, Jupyter, SnapML, dmlc XGBoost, LightGBM, Keras, TensorFlow, PyTorch, Chainer, MATLAB, SAS
- Middle Section – Model Conversion and Optimization

  Arrows indicate the flow of trained models into intermediate formats and optimization tools:

  - Model Formats/Tools: ONNX, PyTorch, IBM SnapML, Deep Learning Compiler (DLC), watsonx.ai® VLLM
- Right Section – Deploy AI at Scale

  This section shows deployment on IBM z17 and LinuxONE 5 platforms, with the following key focus areas:

  - Applications:
    - Industries: Banking, Finance, Insurance, Retail, Hospitality, Transport, Healthcare, Government
  - Offerings:
    - Machine Learning for z/OS, Cloud Pak for Data
    - Db2 with SQL Data Insights, Accelerator for z/OS
    - Red Hat OpenShift AI, watsonx.ai
  - IT Operations:
    - Db2 AI for z/OS, IBM Connect for z/OS
    - INSTANA, Cloud Pak for AIOps
  - Frameworks:
    - TensorFlow, Keras, Spark, Scikit-learn, Anaconda, ONNX

- Libraries/Compilers:
  - Python, Go, GCC, C++, Fortran, Java, Node.js, Rust, Perl, Ruby, R, PHP, Swift, Lua, Julia
- Operating Systems/Virtualization:
  - z/OS, Linux, Red Hat Enterprise Linux, Ubuntu, SUSE

2

Entry points to AI on IBM Z

As more organizations look to integrate AI into heir applications and processes, it is important to know that there is no single way in getting started. Integrating AI into their operations has been a critical enabler of operational efficiency and innovation. Depending on business goals, the technical expertise the team has and how quickly the organization wants to see their return on investments, there are three entry points to practically begin your AI to IBM Z journey.

- The first entry point is ideal for those seeking a quick, low-effort solution—especially if they lack in-house data science expertise. Turnkey solutions are pre-packaged offerings with AI capabilities embedded for specific use cases, enabling rapid deployment and immediate business value with minimal configuration.
- The second entry point is relevant to organizations that have already started their AI journey in the cloud or on-premises and want to reuse existing AI models—whether from IBM or third-party vendors—while taking advantage of IBM Zs powerful inferencing capabilities for efficient, scalable deployment. This approach simplifies integration into your z/OS workflows and delivers performance optimizations. It is also ideal for leveraging existing Large Language Models (LLMs), which you can fine-tune before deploying on IBM Z, where hardware acceleration supports model deployment and inference. This option offers optimized time to value and requires only moderate AI skills.
- The third entry point is for organizations with in-house data science skills and want full control over model development. Building your own custom models gives you the most flexibility. This path requires the right skills to cleanse data, train models, and test them, with additional fine-tuning for customization. Your investment in skills and tools creates a strong foundation for implementing future AI use cases.

Regardless of the entry point, every successful AI journey begins with clearly defined use cases and well-articulated business goals. With these foundations in place, AI-powered applications can deliver actionable insights that drive more insightful decision making and better business outcomes.

An analogy with eating pizza helps illustrate the three entry points shown in Figure 2-1. Imagine you are at home craving pizza. You have three options:

© Copyright IBM Corp. 2025.

- Turnkey solution: Call a restaurant and have a ready-made pizza delivered. This represents a fully packaged AI solution—quick to implement, with immediate value and no need for AI or data science skills to build a model.
- Select a model: Visit a supermarket and buy a pre-made pizza. It still requires some preparation at home before you can eat it. This is similar to selecting a model from IBM or a vendor, which you then deploy and integrate into your environment. This approach accelerates time to value and leverages existing AI and domain expertise.
- Build your own model: With the right expertise, ingredients, and tools, you can make a pizza from scratch. This gives you full control over the ingredients, shape, and flavor—whether circular, rectangular, or uniquely customized. Similarly, building an AI model end-to-end offers maximum flexibility and customization tailored to your organization's needs.

This analogy mirrors the different ways AI can be implemented—from ready-to-use solutions to fully customized deployments.

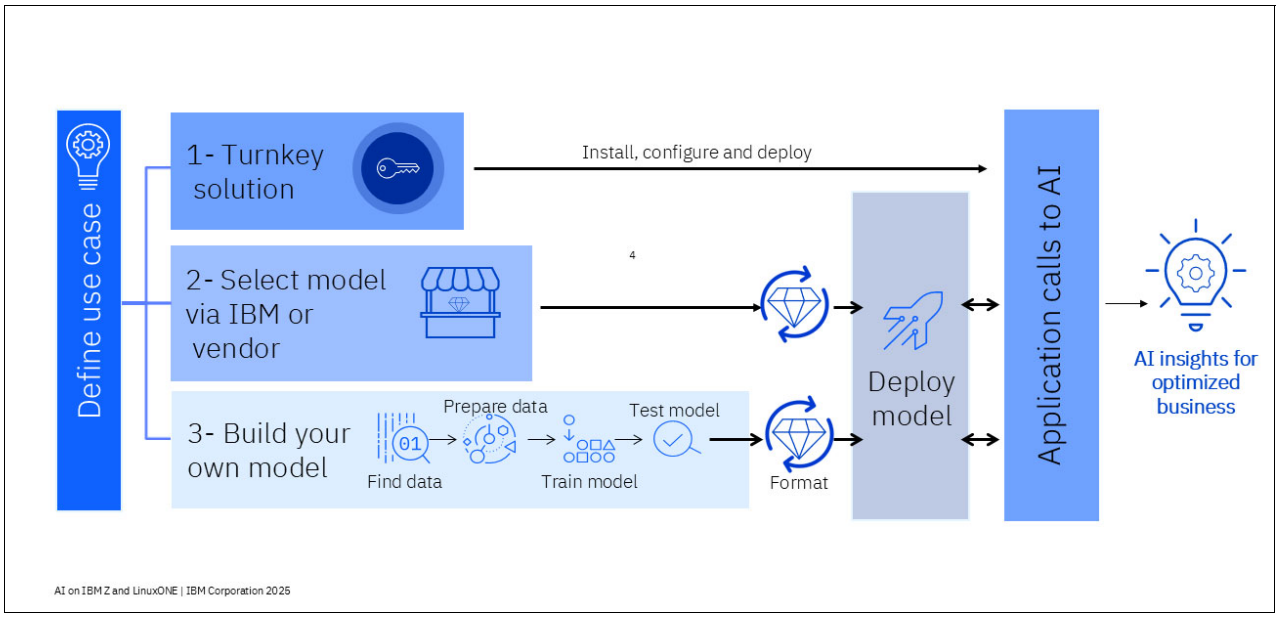


Figure 2-1 Entry points to AI on IBM Z

2.1 The AI life cycle methodology

The AI lifecycle outlines the end-to-end process of developing, deploying, and maintaining AI solutions—from identifying business problems and preparing data, to model development, deployment, monitoring, and refinement. Understanding this lifecycle will help you to :

- Align AI initiatives with business goals.
- Identify where you currently are in your AI journey.
- Choose the most appropriate entry point based on your capabilities and needs.

Without this context, entry points may seem like isolated options rather than parts of a cohesive strategy.

The AI Lifecycle Methodology consists of 7 important steps that helps to ensure that goals can be achieved. An overview of the AI lifecycle methodology is shown in Figure 2-2.

01	02	03	04	05	06	07
Define *your project goals*	**Choose** *a tool*	**Prepare** *the data*	**Train** *your model*	**Deploy** *your model*	**Serve** *model for inference*	**AI Governance** *monitor your model*
What do you want to find out? Do you have the data to analyze?	Pick the tools and entry point that matches your data and desired outcome	Refine the data Add the data as a project asset or in a data repository	Train the model with the data you supply Let a model building tool choose estimators and optimize or choose your own	Move your model from development environment to a production environment	Make a deployed model accessible and usable for real time predictions or inference Exposing a model such as through a REST API	Monitor your model to evaluate the performance of models in production environments for bias and drift Provide explainability of features contributing to a prediction
Key Persona(s): - Line of Business - Data Scientist **Stakeholder:** - IT Architect - Application Architect **Influencer:** - Data Engineer	**Key Persona(s):** - Data Scientist **Stakeholder:** - IT Architect - Application Architect **Influencer:** - Line of Business - Data Engineer	**Key Persona(s):** - Data Engineer **Stakeholder:** - Data Owners - Data Scientist **Influencer:** - Line of Business - Data Wrangler	**Key Persona(s):** - Data scientist **Stakeholder:** - IT Architect - Application Architect **Influencer:** - Data science manager	**Key Persona(s):** - IT Architect - Application Architect **Stakeholder:** - Data Scientist - ML Engineer **Influencer:** - Line of Business	**Key Persona(s):** - ML Engineer - Data Scientist **Stakeholder:** - Data Scientist - ML Engineer **Influencer:** - Line of Business	**Key Persona(s):** - ML Engineer - Data Scientist - Data Engineer **Stakeholder:** - Data Scientist - ML Engineer **Influencer:** - Data Engineer

Figure 2-2 AI lifecycle methodology

This methodology applies especially to predictive AI that leverages Machine Learning and Deep Learning models. If your use case involves large language models (LLMs), you do not need to build them from scratch because the time and resource requirements are significant. You can leverage existing LLMs and fine-tune them with your data or use prompt engineering to adapt the model for a specific task. The model can then be deployed, served and governed as described in the methodology table.

Understanding the AI life cycle can help you to match entry points to your organization's readiness and roles.

Each entry point corresponds to different stages and roles in the AI lifecycle:

- Turnkey solutions align with organizations focused on deployment and quick wins.
- Model integration fits those in the deployment and scaling phase.
- Custom model development is for teams engaged in data preparation, training, and experimentation.

By first understanding the lifecycle, you will be able to better assess your organization's readiness, identify gaps in skills or infrastructure, and make informed decisions about where to begin or expand.

2.2 AI on IBM Z products used in the AI lifecycle

Based on the described lifecycle, you can identify which tools support each phase of the AI model lifecycle. Figure 2-3 provides an overview of tools and frameworks in a matrix that maps each AI component in the IBM Z software stack to its optimal use case and supported operating systems. Some offerings, such as Machine Learning for IBM z/OS, can be used in multiple phases. A star highlights the phase or phases where each offering is primarily used.

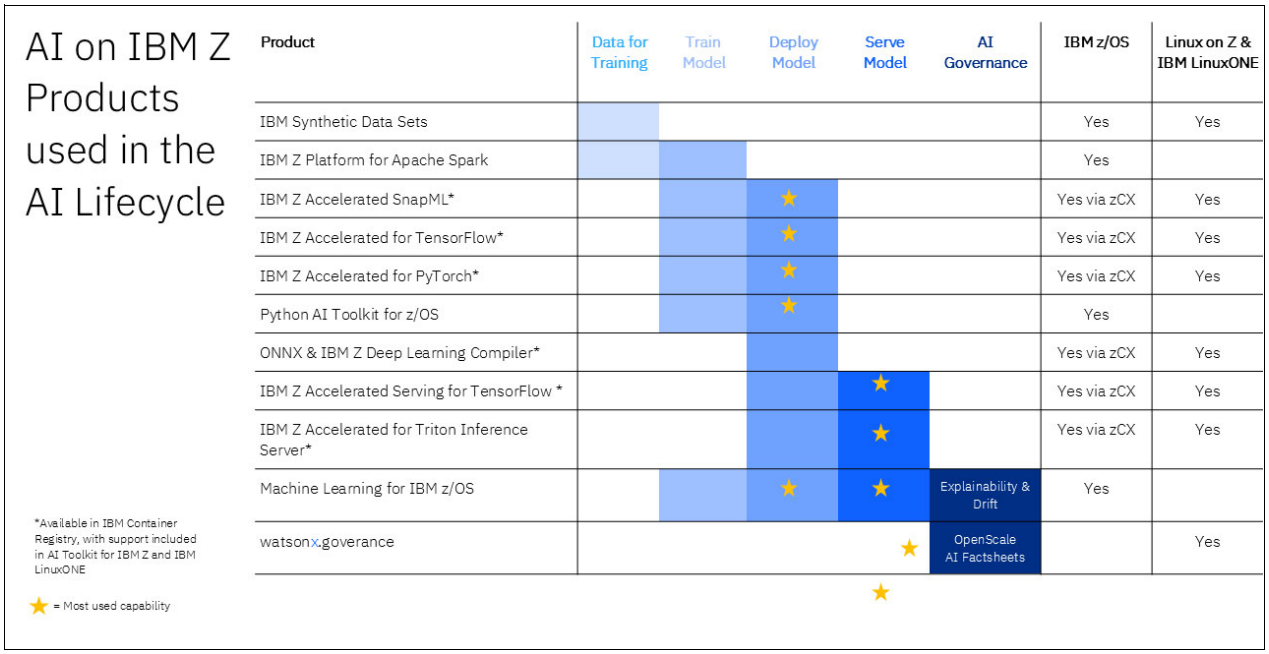


Figure 2-3 AI on IBM Z software stack mapping to the AI lifecycle

In the next chapters, we will present use case examples based for each entry point. The use cases will highlight selected tools and frameworks from the Z software stack and the AI on IBM Z portfolio.

3

Entry Point 1 - IBM turnkey solution examples

This chapter presents a use case example that leverages turnkey solutions (Figure 3-1) to show how you can deploy pre-built AI models and applications already integrated into existing or new offerings from IBM or Integrated Software Vendors (ISVs).

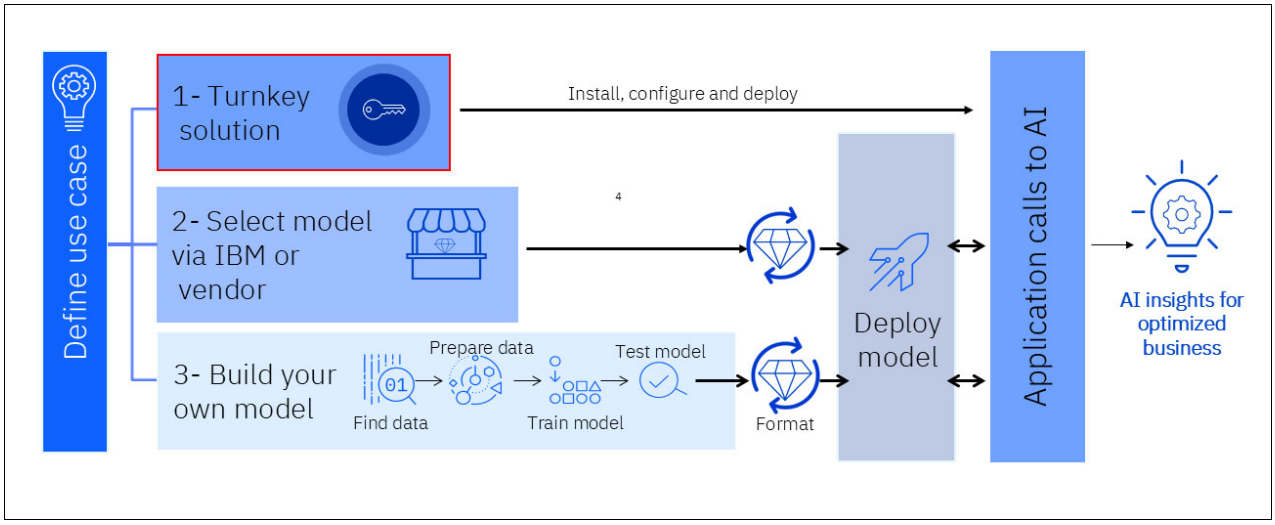


Figure 3-1 AI Entry Point 2 - Turnkey solutions

© Copyright IBM Corp. 2025.

3.1 AI-enhanced SQL analytics - SQL Data Insights in Db2 for z/OS v13

This section focuses on how SQL Data Insights (SQL DI) uses AI to enhance database queries and deliver advanced analytics without requiring data movement or deep data science expertise.

3.1.1 Use case objectives

Client implementations often serve as useful examples. This example is based on a project at a large insurance provider. This insurance provider intends to leverage AI to offer personalized services as a key differentiator and improve the likelihood that insurance offers convert into contracts, either in person through a sales representative or online by the end user. They also aim to reduce manual work and improve response times. They are exploring how to achieve this by applying AI to client data that resides on Db2 for z/OS to unlock hidden patterns. One key requirement is to implement a high-performance solution based on AI predictions with high accuracy.

3.1.2 Challenges

Db2 for z/OS data must remain in the secure z/OS environment to meet data privacy and security requirements. The organization also faces a challenge with resource and skills availability. The Db2 for z/OS team has limited data science and AI skills, and the application team is not familiar with the IBM Z platform.

3.1.3 Solution

To overcome these challenges, the insurance company can use a turnkey solution called SQL Data Insights. This solution helps the insurance company accelerate time to value. SQL Data Insights provides embedded AI capabilities with IBM Db2 for z/OS v13. It uses a self-supervised vector embedding approach that does not require labeling or data pre-processing. This approach produces a generalized AI model. The model, often called a Large Database Model, is a powerful deep learning model designed to answer several questions derived from the data.

The insurance company uses functions such as similarity, dissimilarity, and clustering with simple SQL statements. These functions compare in-progress quotes with similar historical quotes and provide the user with a real-time conversion probability score. SQL Data Insights enables the insurance company to run highly performant and accurate AI-enhanced semantic queries on demand. These queries support applications such as product recommendations, personalized experiences through customer segmentation, and insurance fraud detection.

The variety of implemented use cases was possible because of the accelerated time to value. The insurance company turned the idea into a successful production implementation within 4 hours. Typically, an AI implementation takes several months or even years to transform a use case idea into an operational AI deployment.

The client achieved this quick outcome by using the embedded AI capabilities in IBM Db2 for z/OS. They leveraged AI-infused semantic queries with powerful deep learning model inferencing through easy-to-use, standard SQL interfaces. This approach minimizes developer complexity and eliminates the need for data science skills.

From a business perspective, they increase the quote-to-conversion rate and achieve a significant sales boost. This outcome results from the solution's high accuracy, delivering 94% prediction accuracy in near real time.

AI capabilities in Db2 for z/OS eliminate data movement outside the platform and reduce security risks. Using SQL Data Insights as a turnkey solution helps you avoid costly data extraction and preparation stages that are usually required for model training.

Further discussion on this particular use case from IBM Z Day 2025 SE can be found at: https://mediacenter.ibm.com/media/1_2g9nvgrf

3.1.4 Additional use cases with SQL Data Insights

Additional use cases can range from finding entities similar to those identified to identifying IT anomalies within your system data. One of the key strengths of AI embedded in Db2 for z/OS is the combination of the Db2 built-in functions for SQL DI with other SQL operators.

There are no pre-defined use cases for SQL DI, which can be adapted without changes for every client infrastructure. Finding use cases for SQL DI and AI on IBM Z in general is an effort, which needs close collaboration from the IBM Z infrastructure department and the Data Analytics department.

As described in the use case in section 3.1.1, "Use case objectives" on page 30, the value of SQL DI can be summarized as follows:

- A self-supervised approach that does not require specialized data science skills.
- No data movement is required because model training and inferencing are performed within the system. This enables AI in a secure, in-database environment.
- A single model is used for various Db2 built-in functions for SQL DI.
- Standard SQL is used to call Db2 built-in functions for SQL DI, so you can call them from any SQL client.
- IBM Z hardware acceleration is used, including the Integrated AI Accelerator in Telum and Telum II.

For getting started with SQL DI and AI on IBM Z in general, an AI on IBM Z and LinuxONE Discovery Workshop is recommended. For more information on how to get started please refer to chapter 5. Please feel free to also reach out to your IBM Representative or the AI on IBM Z and LinuxONE team directly at aionz@us.ibm.com.

3.1.5 How it works

SQL Data Insights combines built-in Db2 functions with an AI-driven architecture to deliver semantic query capabilities directly in Db2 for z/OS. This section explains the available functions, the underlying architecture, and best practices for effective use.

SQL Data Insights (SQL DI) is a component of Db2 for z/OS version 13 and can be ordered no-charge via ShopZ with the FMID HDBDD18. It enables the execution of AI-infused SQL-based semantic queries, which can be called from any SQL client.

SQL DI is a crucial component of the business insights aspect of the AI on IBM Z ecosystem. Its lightweight installation and setup, coupled with the advantage to create reliable results without deeper user insights of statistical algorithms, make SQL DI an excellent entry point into AI on IBM Z. Db2 built-in functions for SQL DI can be combined with SQL operators.

Target groups are not only business users and application developers, but also database administrators, who can now offer AI capabilities directly on Db2 for z/OS. Offloading sensitive data to identify interdependencies and relations is therefore not needed.

Db2 built-in functions

SQL Data Insights provides several AI-infused functions that you can call from any SQL client using standard syntax. These functions enable similarity searches, clustering, analogies, and pattern detection without requiring data movement or specialized AI skills.

At the time of this IBM Redbooks publication, the Db2 built-in functions that are available for SQL DI are shown in Table 3-1. For more information about these functions, see Db2 built-in functions for SQL DI.

Table 3-1 Db2 built-in functions for SQL DI

Function name	Description and example

AI_SIMILARITY	Based on the given attributes of a given entity, find similar or dissimilar records of entities.
	
	Use AI_SIMILARITY to identify the top five customers by ID most similar to the customer with ID '1602-KBYPQ', who churned. To avoid further churn, you can target specialized marketing actions to these customers.
	Example SQL query:
	`SELECT AI_SIMILARITY(CUSTOMERID, '1602-KBYPQ'), CHURN.*`
	`FROM CHURN`
	`ORDER BY 1 DESC`
	`FETCH FIRST 5 ROWS ONLY;`
	
	Tip: Include CHURN.* to return the original columns together with the similarity score. For readability, alias the function result (for example, AS SIMILARITY_SCORE) and reference that alias in ORDER BY.

AI_SEMANTIC_ CLUSTER	Based on the given attributes of a group of entities, find similar or dissimilar records of entities.
	
	In this example, not only the customer with ID '1602-KBYPQ' churned, but also two other customers with IDs '1503-BKQPY' and '1702-YQKBP'. These three customers form a cluster. With AI_SEMANTIC_CLUSTER, you can find the top five customers that are close to this cluster. You can then target specialized marketing actions to these customers.
	
	Example SQL query:
	`SELECT AI_SEMANTIC_CLUSTER(CUSTOMERID, '1602-KBYPQ', '1503-BKQPY',`
	`'1702-YQKBP'), CHURN.*`
	`FROM CHURN`
	`ORDER BY 1 DESC`
	`FETCH FIRST 5 ROWS ONLY;`
	
	Tip: Use 'CHURN.*' in SQL to not only get the AI_SEMANTIC_CLUSTER score as a result.

Function name	Description and example

AI_ANALOGY	Based on the relationship between a pair of entities, you can apply this relationship to a second pair. For example: A is to B as C is to “?”. The question mark represents the result of the function.
	
	In this example, you have a list of music songs with metadata, and you want to find the appropriate music genre for a specific artist. The genre of the artist 'ZZ Top' is defined as 'Blues'. Use AI_ANALOGY to find the genre of the artist 'Daft Punk'.
	
	Example SQL query:
	`SELECT AI_ANALOGY('ZZ Top' USING MODEL COLUMN ARTIST_NAME,`
	`'Blues' USING MODEL COLUMN MUSIC_GENRE,`
	`'Daft Punk' USING MODEL COLUMN ARTIST_NAME,`
	`MUSIC_GENRE) AS DECIMAL(3 , 2)`
	`FROM MUSIC_SONGS_DATABASE`
	`ORDER BY 1 DESC`
	`FETCH FIRST 1 ROWS ONLY;`

AI_COMMONALITY	Based on a given dataset, find common patterns or outliers.
	
	You can clean up a playlist of music songs with metadata by removing outliers. Use AI_COMMONALITY to identify these outliers.
	Example SQL query:
	`SELECT AI_COMMONALITY(MUSIC_ID) AS SCORE, C.*`
	`FROM MUSIC_SONGS_PLAYLIST C`
	`ORDER BY SCORE ASC; - use ASC if lower scores indicate outliers`
	
	Tip: Return the original playlist columns together with the commonality score by selecting C.* alongside AI_COMMONALITY(...) (aliased, for example, as COMMONALITY_SCORE). Use the alias in ORDER BY for readability instead of positional ordering.

Architecture and usage flow

SQL Data Insights uses a lightweight Spark-based service for initial model training and Db2 for z/OS for inference.

- Training phase: You select data types and key columns via the browser UI, REST APIs, or CLI. The service creates embeddings and loads the model table into Db2.
- Inference phase: Any SQL client can call AI functions against Db2 tables without invoking Spark.

The architecture leverages zIIP offload and optimized libraries (zAIO and OpenBLAS) for performance. Retraining is recommended when source data changes significantly.

SQL DI is lightweight to install because it is based on a Spark service for training the AI model. Figure 3-2 shows the SQL DI architecture from two perspectives: training and semantic queries. In this section, we explain both perspectives.

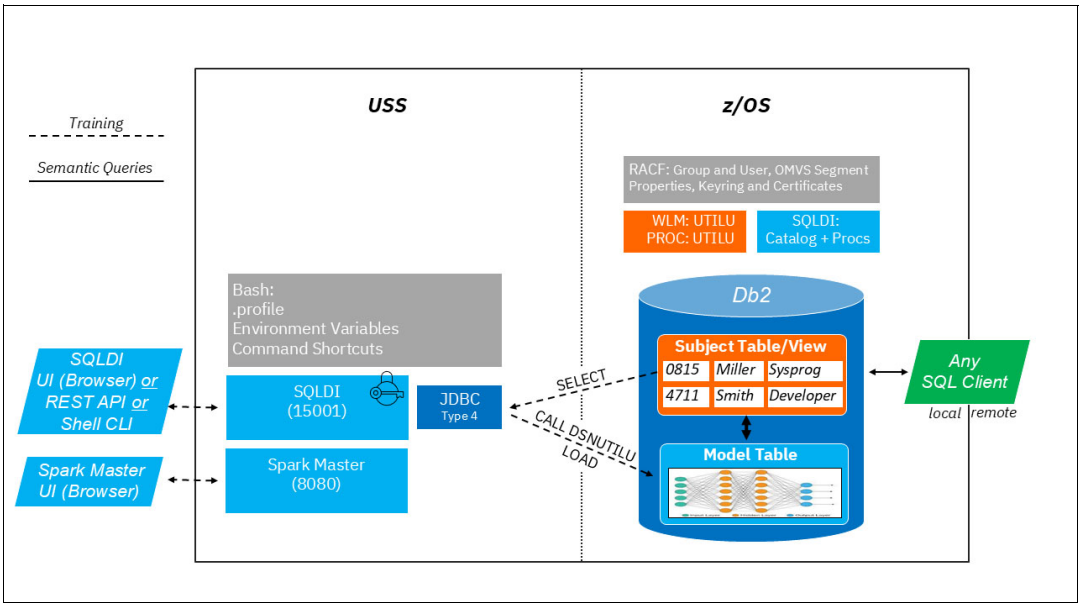


Figure 3-2 SQL Data Insights architecture

You can perform the initial training of the AI model, which is used for all SQL DI functions, by using the SQL DI browser UI, RESTful APIs, or shell CLIs. During this process, you can select data types and define the key column if it is not detected automatically. The SQL DI service uses a JDBC connection to select the data (Subject Table/View in Figure 3-2), trains an AI model (shown with dashed lines in Figure 3-2) and loads the model table into Db2 for z/OS (Model Table in Figure 3-2).

If the associated data (Subject Table/View in Figure 3-2), is changed, a retraining is recommended. If not, semantic queries can only run against data, which were be represented at the time of the training of the AI model. SQL DI is offering ways for a non-disruptive training.[1] SLQ DIs trainings and UI services are zIIP-enabled.[2]

Any SQL client can connect to Db2 for z/OS and call the Db2 built-in functions that are described in Table 3-1 when using an Ai-infused SQL query. Db2 for z/OS uses the model table (shown with a solid line in Figure 3-2) and does not rely on Spark training services. SQL DI leverages the zAIO and OpenBLAS libraries. The IBM Z Artificial Intelligence Optimization Library and the OpenBLAS library provide core functions for database embeddings and algebra operations.

3.1.6 Best practices for using SQL Data Insights

To effectively use SQL DI, identify impactful use cases in collaboration with data analytics and business stakeholders. Start with a small dataset to understand the infrastructure, plan capacity, and measure model training times and other factors.

Exclude columns with poor data quality, such as those with many NULL values. Preparing the data by creating a view as input for model training can be helpful. Reassign data types during the training phase if necessary, and exclude columns such as LOBs or long VARCHAR from SQL DI usage.

[1] APAR PH60870, released in May 2024, is delivering this function.See New model retraining capability of SQL Data Insights

[2] See Chapter 1. Overview of SQL Data Insights, page 1 in the SQL DI user guide

To maximize accuracy and efficiency:

- Start with a small dataset to validate infrastructure and measure training time.
- Exclude columns with poor data quality or high NULL values.
- Prepare input views for model training and reassign data types if needed.
- Avoid using LOBs or very long VARCHAR columns in SQL DI queries.
- Plan for periodic retraining when data evolves.

3.2 Intelligent Db2 operations - Db2 AI for z/OS

Operating Db2 for z/OS efficiently requires specialized skills, which can be challenging to maintain as workloads grow and experienced professionals retire. Db2 AI for z/OS (Db2ZAI) addresses these challenges by embedding machine learning into Db2 operations. This section explains its key capabilities—SQL optimization, system assessment, and distributed connection control—and provides guidance for effective use.

IBM Db2 AI for z/OS (Db2ZAI) is powered by AI and provides three key features that demonstrate intelligent infrastructure:

- SQL Optimization: Db2ZAI empowers the optimizer in your Db2 for z/OS engine to determine the best-performing query access paths based on your workload characteristics.
- System Assessment and Performance Insights: Db2ZAI detects system-level and application-level performance exceptions and provides recommended insights and actions for tuning based on your environment.
- Distributed Connection Control: Db2ZAI protects Db2 for z/OS from unintentional disruptions and intrusions that can originate from remote applications.

For additional, general information on Db2 AI for z/OS, see IBM Db2 AI for z/OS.

3.2.1 SQL optimization

Db2ZAI enhances the Db2 optimizer by learning from your workload history to recommend the most efficient query access paths. Db2ZAI collects data from the optimizer and query execution history, trains a model to identify patterns, and deploys the model to score incoming SQL statements. The benefits of using Db2ZAI include improved query performance, reduced CPU usage, and optimized parallelism for static and dynamic SQL.

Tip: Use Db2ZAI scoring during the bind or prepare phase to ensure optimal access paths without degrading performance.

The optimizer consists of Relational Data Services (RDS) components that govern query transformation, access path selection, run time, and parallelism for every SQL statement in your system. The access path for an SQL statement determines how Db2 accesses the data that the query specifies. It defines the indexes and tables that are accessed, the access methods that are used, and the order in which objects are accessed.

Db2ZAI uses machine learning technology to collect data from the Db2 for z/OS optimizer and query execution history, which are derived from workloads in your operating environment. During model training, Db2ZAI identifies patterns in this data and learns the optimal access paths for queries entering Db2 for z/OS. After training, the model is deployed into production

to provide insights for the optimizer's access path selection. Figure 2 shows the architecture and workflow.

Leveraging machine learning technology, Db2ZAI collects data from the Db2 for z/OS optimizer and the query execution history, which are derived from workloads in your unique operating environment. As part of the model training process, Db2ZAI then finds patterns from this data and learns the optimal access paths for queries entering Db2 for z/OS. Once trained, the model is ready to be deployed into production, providing insights to the optimizer's access path selection. Figure 3-3 shows the described architecture and workflow.

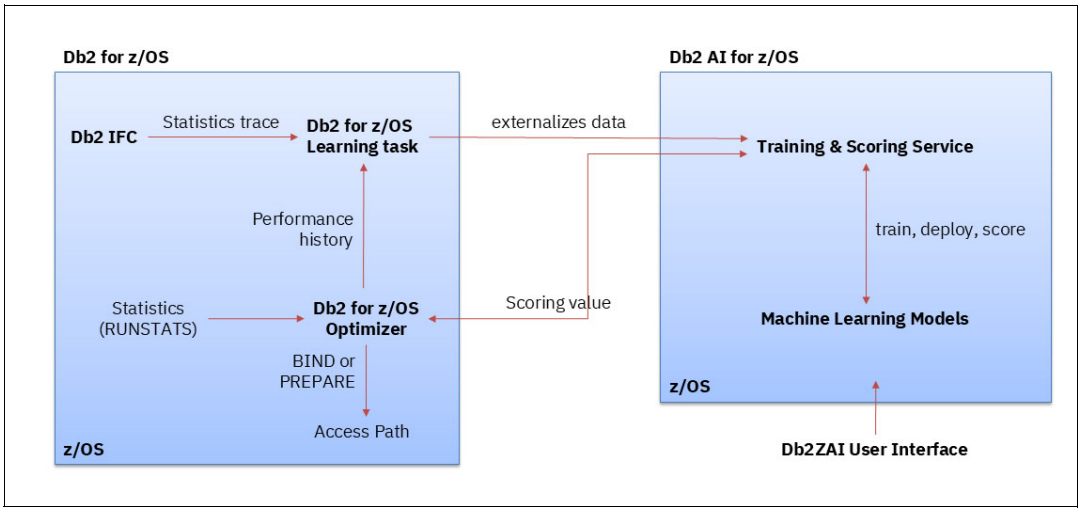


Figure 3-3 Architecture and workflow for SQL performance optimization

The AI-extracted insights augment the optimizer's ability to select the best query access path. This information is unique to your environment and is not available to the traditional query optimizer.

Db2ZAI provides additional capabilities, including:

- Predicting host variables for a given query by analyzing patterns from repeated evaluations of associated data for similar queries.
- Predicting the number of rows in the result set that the application will actually fetch.
- Estimating the benefit of parallel processing for static and dynamic SQL more accurately, which can improve elapsed time and reduce chargeable CPU.
- Determining whether sort processes can benefit from increased memory utilization.

With the intelligence gained from these models, the query optimizer is better prepared to identify optimal access paths for SQL statements. During the bind or prepare phase, the Db2 for z/OS optimizer can use the optional scoring value provided by Db2ZAI. The mechanism within Db2ZAI ensures that access paths are not degraded.

Internal IBM benchmarks show performance benefits, including reduced CPU usage and shorter elapsed times for these types of queries. In addition, Db2ZAI enables the workload associated with the models to run on the IBM Z Integrated Information Processor (zIIP), minimizing the impact on general processor CPU.

3.2.2 System Assessment and Performance Insights

Db2ZAI addresses the challenges associated with analyzing the vast amounts of performance data generated by Db2 for z/OS. System programmers and performance analysts often face constraints in time and expertise, which can limit their ability to interpret metrics and proactively tune the system. Db2ZAI applies machine learning techniques to identify anomalies in system performance after they occur. It then provides a set of recommended actions to help mitigate the impact of these events and improve overall system efficiency.

Db2ZAI applies machine learning to analyze system-level and application-level performance metrics.

Figure 3-4 shows a system assessment report from a test system, which clients can schedule to run at a regular time or on demand. The report can include insights like issues with metrics that are listed in different categories. Performance Insights is showing similar data, but primarily focusing on the presentation of performance data.

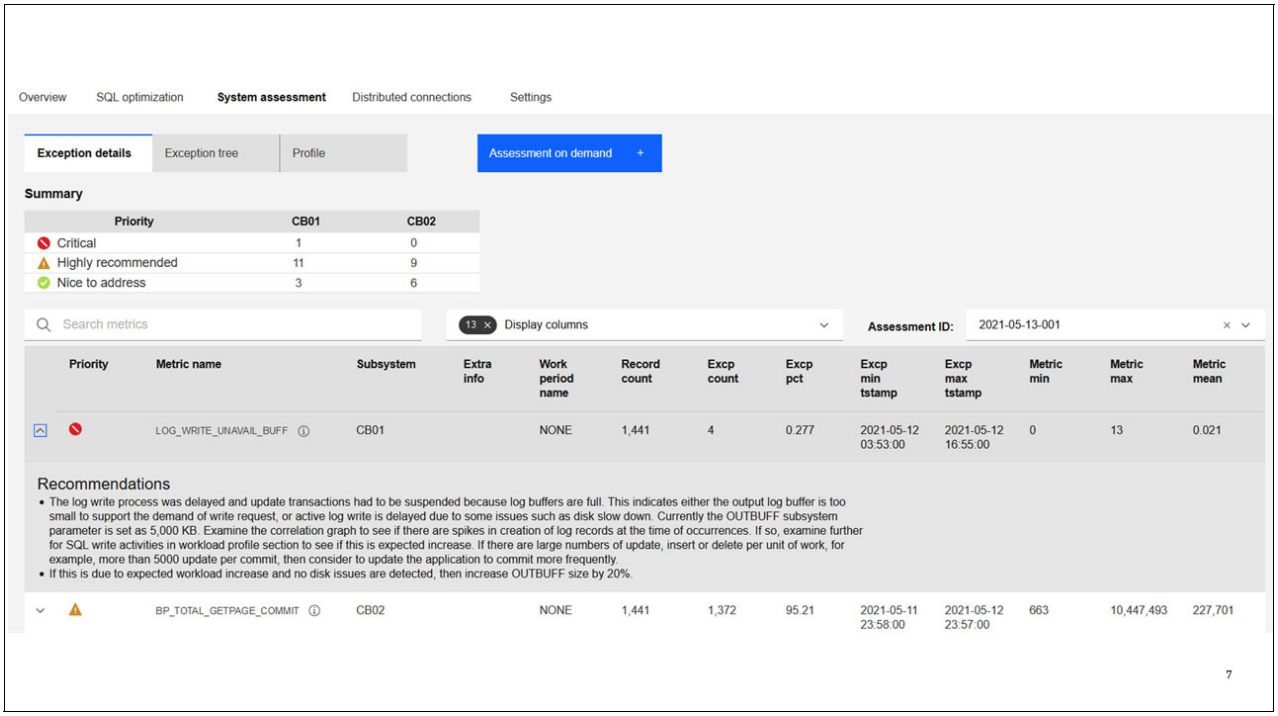


Figure 3-4 Overview of the System Assessment UI - Summary of a System Assessment report

You can drill down into specific metrics to view detailed information about an issue. Figure 3-5 shows a detailed view of the critical metric from Figure 3-4, which you can view interactively. The right side of the figure displays related metrics.

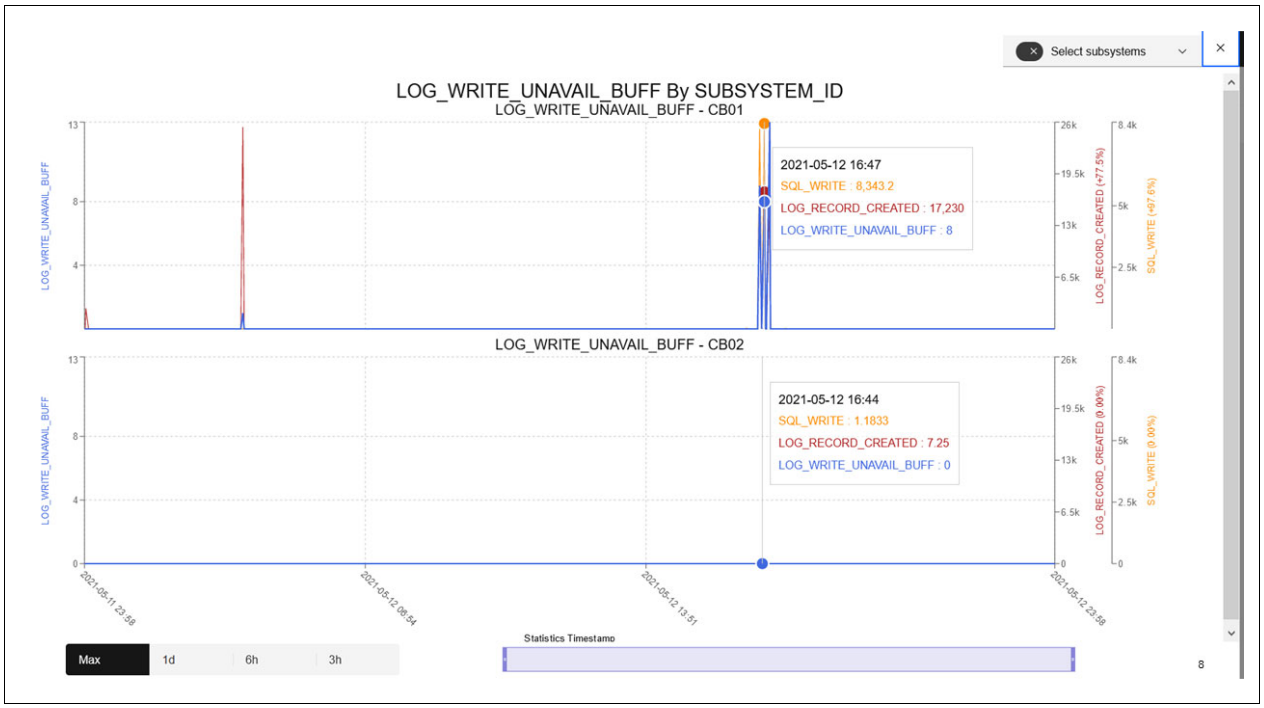


Figure 3-5 Overview of the System Assessment UI - further analysis of an Db2 for z/OS metric

Important: The System Assessment feature of Db2ZAI is not a replacement for a real-time monitor.

The screenshots in this section are taken from a demonstration environment. In a production environment, you might see different results. You might also want to set up work periods to differentiate workloads. For more information, see the Db2ZAI documentation:

- Setting up Db2ZAI Db2 system assessments
- Evaluating Db2 system performance using system assessments

With Db2ZAI 1.6.2, imported data system assessment and flexible system assessment were introduced. For more information, see the Db2ZAI documentation:

- What's new in Db2 AI for z/OS 1.6.x
- Types of Db2ZAI connections

Figure 3-6 shows the system assessment architecture with the new imported data system assessment function.

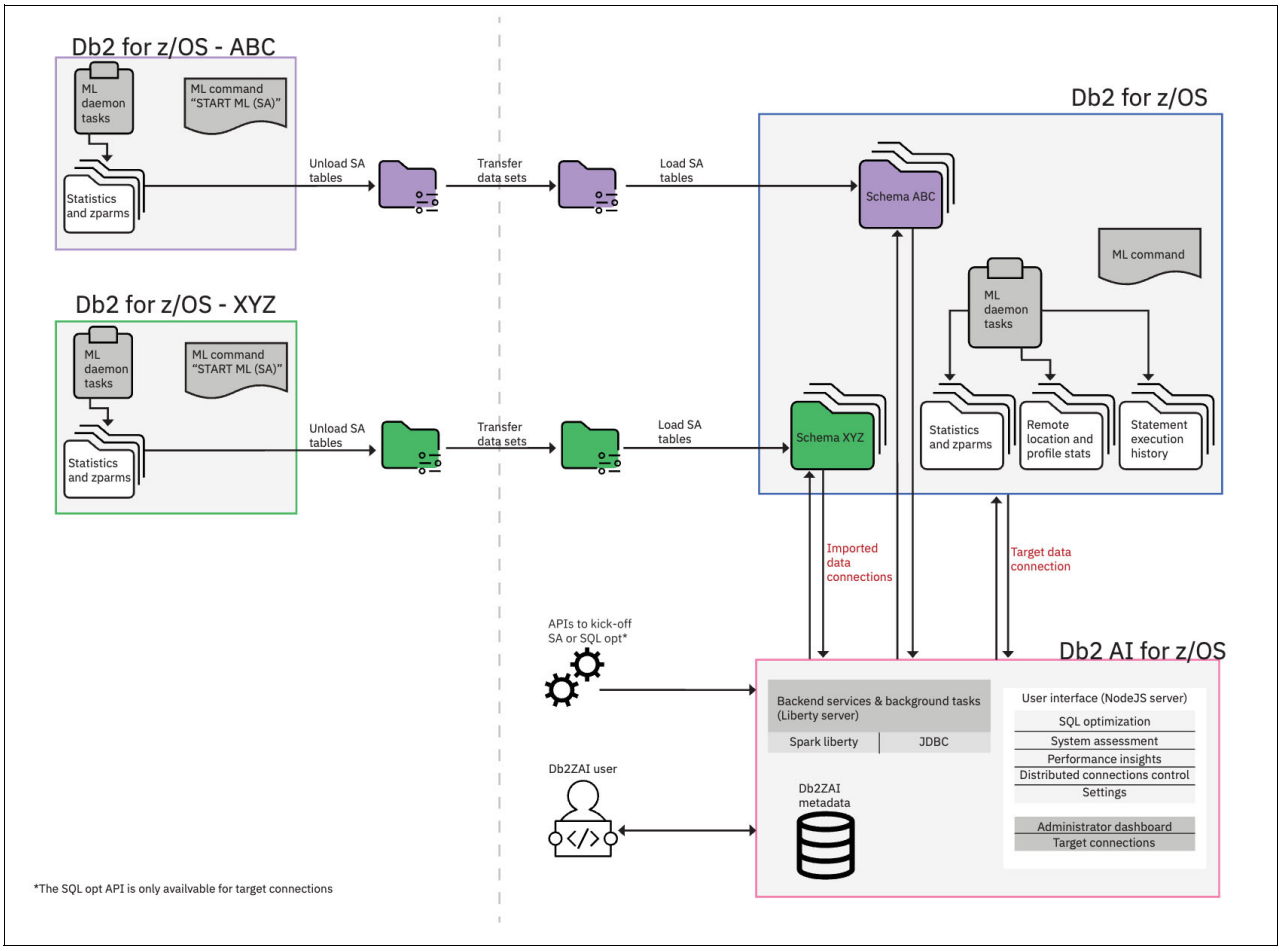


Figure 3-6 System Assessment architecture

3.2.3 Distributed connection control

Db2ZAI monitors and controls remote application connections to protect subsystem stability. Distributed connection control records and then analyzes remote access of applications against Db2 subsystems and produces connection and thread profiles that allow the administrator to control warning and exception levels for remote connections, and similarly, warning and exception levels for threads driven by remote connections. These profiles help Db2 protect the subsystem from remote database applications that may misbehave, for whatever reason. This helps to preserve the stability and performance of your database subsystems.

Distributed Connection Control also supports in identifying the applications accessing Db2 for z/OS including the driver level. These information are useful for identifying the source of distributed connection spikes and issues.

Figure 3-7 shows the flow performed by the Distributed Connection Control feature.

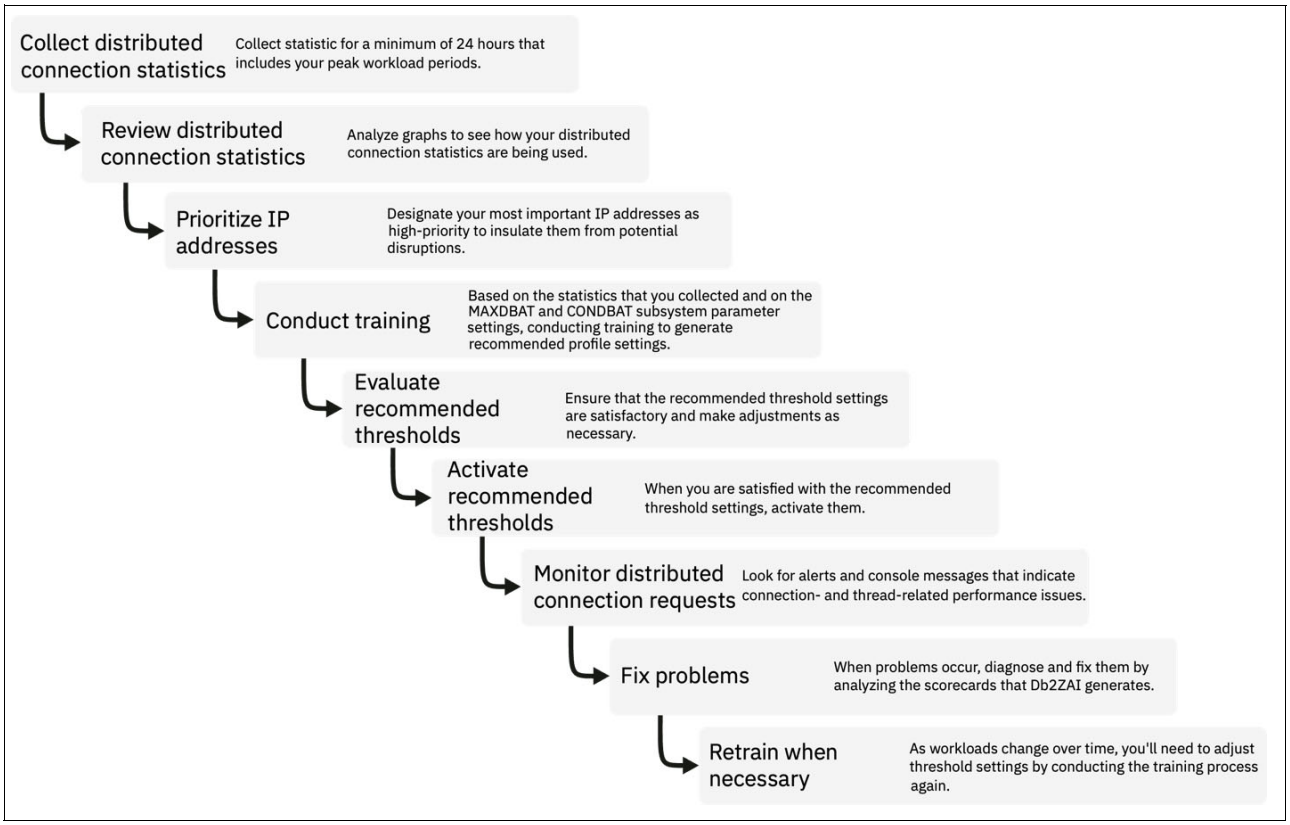


Figure 3-7 Distributed connection control task flow

3.2.4 Best practices for effective use

To maximize the value of Db2ZAI:

- Enable zIIP offload for model training and scoring to minimize CPU impact.
- Schedule regular system assessments to proactively detect anomalies.
- Review optimizer recommendations after major workload changes or application updates.
- Adjust distributed connection thresholds based on observed traffic patterns.
- Keep Db2ZAI models updated by retraining when workload characteristics evolve.

3.3 Operations productivity and skills acceleration - watsonx Assistant for Z

Managing IBM Z operations requires specialized expertise, and onboarding new professionals can be time-consuming. As workloads grow and experienced staff retire, organizations face skill gaps that impact efficiency and resilience. IBM watsonx Assistant for Z (WA4Z) addresses these challenges by combining generative AI with automation to simplify tasks, accelerate learning, and boost productivity.

Reducing the learning curve and boosting productivity in IT operations is critical for organizations that depend on IBM Z. As workloads grow in complexity and business demands increase, IT teams need efficient ways to access information, resolve issues quickly, and maintain system stability. Intelligent tools and automation can help address these needs by

simplifying tasks and providing actionable insights. In this section, we describe the challenge and present a solution.

3.3.1 Challenge

Early-tenure professionals often struggle to find accurate information quickly, navigate multiple tools, and perform tasks confidently in production environments. These challenges lead to delays, frequent interruptions of subject matter experts, and increased risk of errors. Organizations need a solution that reduces dependency on scarce skills while maintaining operational stability.

Figure 3-8 illustrates the on-boarding journey for early-tenure IBM Z professionals and the challenges they often face. For example, finding relevant information to remediate a problem can be difficult because the information is spread across multiple sources. Early-tenure professionals frequently seek help from mentors and experts, which can lead to constant interruptions of the experts' tasks—often with the same question from different colleagues. At times, IBM Z professionals might receive conflicting advice on how to complete an activity, creating confusion and uncertainty.

Even simple tasks can take a long time at the beginning because users are not familiar with the tools. Before starting, delays can occur while obtaining the correct security permissions. Switching between multiple tools and screens can also be confusing.

When performing tasks for the first time, there is often a fear of causing errors that could impact application availability. Working in production environments is risky, and accidentally bringing down a system can affect business operations.

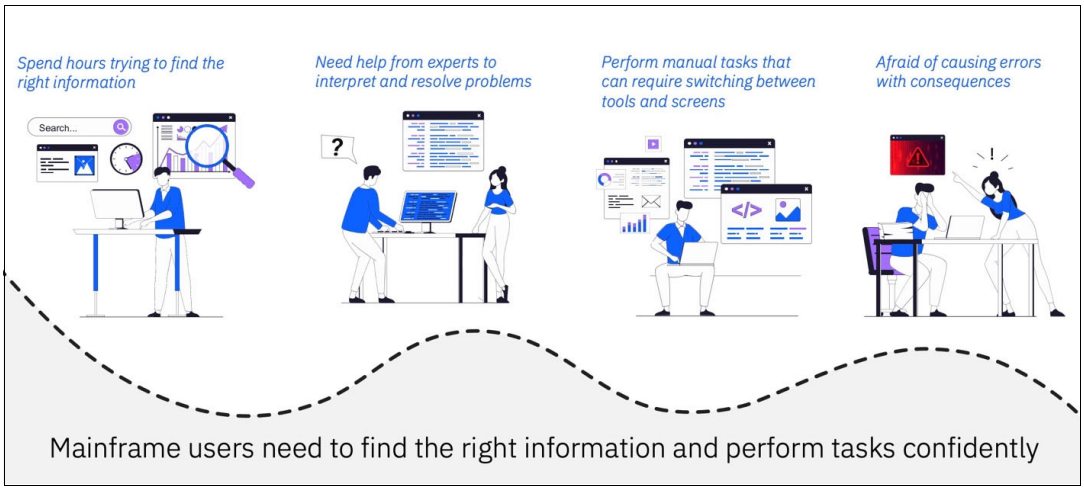


Figure 3-8 User journey

To address these issues, generative AI (GenAI), combined with automation can help tackle the skill challenges and streamline how system programmers, IT operators, and developers acquire knowledge enhance productivity, boost efficiency, and optimize work quality, regardless of their experience level.

3.3.2 Solution

IBM watsonx Assistant for Z provides a single point of access for curated answers to questions related to IBM Z. The assistant puts valuable information at your fingertips with detailed steps and links to explore further, reducing the learning curve and increasing

efficiency. In addition, the AI-powered conversational experience is integrated with automation that simplifies and optimizes the execution of routine operations tasks. Infusing automation into the conversation ensures that common and complex tasks are performed correctly every time, reducing errors and improving productivity. By enabling you to find the right information and perform tasks without interrupting subject matter experts (SMEs), the assistant helps IT operators, system programmers, and other roles build confidence and autonomy.

IBM watsonx Assistant for Z provides a conversational AI experience enriched with IBM Z domain knowledge and integrated automation.

- Generative AI responses with IBM context awareness

  IBM watsonx Assistant for Z provides accurate and current information by leveraging IBM Z domain-specific knowledge. For governance and compliance, the assistant uses the granite-3-8b-instruct LLM model, which meets stringent requirements and is trained on an extensive dataset of more than 12 trillion tokens, including IBM Z-specific content.

  To ground enterprise knowledge with both IBM Z and organizational context, the assistant uses the Retrieval Augmented Generation (RAG) approach. This enables updates to the knowledge base and ensures that responses are current. If the assistant does not know the answer, it does not guess, which helps reduce hallucinations.

  The assistant supports conversational AI in multiple languages in addition to English. The currently supported languages are German, French, Japanese, and Italian. For the most up-to-date list and any newly added languages, see: IBM watsonx Assistant for Z documentation.

- Enriching conversational AI with integrated automation

  Integrating automation with conversational AI simplifies access to live systems, provides operational insights on IBM Z, and reduces the need for specialized skills and tool switching. Automation and AI-driven tools bridge this gap by offering step-by-step assistance, automating repetitive workflows, and delivering real-time recommendations based on conversational context. These capabilities help early-career developers tackle new problems and enable experienced professionals to revisit rarely used skills. By reducing errors, improving efficiency, and minimizing frustration, automation allows mainframe professionals to focus on strategic, high-value initiatives.

  For example, Figure 3-9 illustrates how an early-tenure IT operator can pinpoint performance issues on a z/OS system. Previously, this task required navigating multiple monitoring tools, connecting to different systems, and relying heavily on experts to identify the root cause. With integrated automation, the operator can now gain operational insights in seconds and assess the health of LPARs, Db2 instances, IMS, MQ, CICS subsystems, networks, storage, JVMs, and automated applications that use IBM Z Workload Scheduler and NetView. The operator can also view critical alerts in subsystems such as MQ, CICS, and IMS, and quickly identify problem areas—all through predefined WXA4Z automation.

  The integrated automation uses existing z/OS user security settings when executing automation. It includes a large set of prebuilt skills for common z/OS tasks and AIOps capabilities that accelerate the creation of automation flows.

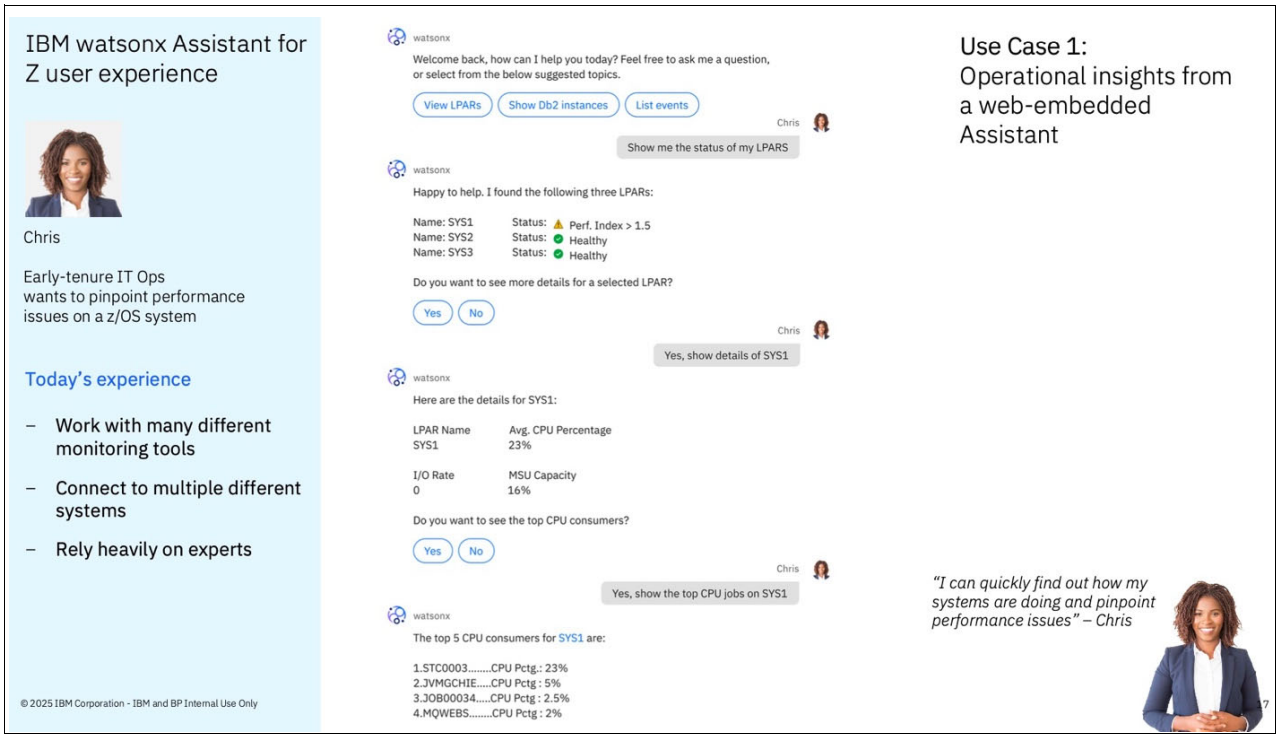


Figure 3-9 Using the assistant to get operational insights

- Configurable assistants

  Creating role-specific assistants ensures that the AI experience aligns with users' responsibilities and skill levels. Organizations can build purpose-built assistants that answer questions and fulfill specific automation requirements, enabling users to complete tasks and complex processes seamlessly within an engaging conversational AI experience. Assistants can be customized by persona, team, or experience level.

  The knowledge base can be enriched with the organization's own documents, including configuration details, problem-resolution procedures, and best practices tailored to the IBM Z environment. This capability is provided by the Bring Your Own Documents (BYOD) feature, which supports multiple formats such as HTML, DOCX, Markdown, Excel, and CSV.

  In addition to prebuilt automation skills, clients can create simple or advanced automations through a low-code approach. During assistant creation and configuration, you can include only the automation that meets the specific business needs of your audience.

- Drive more productivity with Agentic AI

  IBM watsonx Assistant for Z is evolving into an Agentic AI-powered solution that includes specialized IBM Z agents to help you increase productivity.

  During a conversation, you can access specialized AI agents that collaborate to provide unique insights, improve decision-making, and automate multi-step processes with minimal supervision.

  Agentic AI systems represent the next generation of AI. They provide advanced reasoning capabilities, respond to natural language, and make decisions autonomously. AI agents use tools to take automated actions and collaborate with other agents. They plan and act on tasks in response to your questions or prompts. Agents learn iteratively, refining their approaches to better align with defined objectives.

IBM Z agents offer deep domain knowledge to handle complex tasks. They cover both system and middleware, understand unique environment configurations, and include tools that drive automation. You can engage all your AI agents in one place.

With multi-agent orchestration, IBM watsonx Assistant for Z understands deeper user intents and routes requests to the appropriate agent.

3.3.3 Best practices for effective use

To maximize WA4Z benefits:

- Start with high-impact tasks such as performance checks and routine maintenance.
- Use BYOD to integrate organization-specific procedures and best practices.
- Configure role-specific assistants to align with user responsibilities.
- Leverage prebuilt automation skills and extend them with low-code flows for custom tasks.
- Regularly update the knowledge base to ensure responses remain accurate and relevant.

4

Entry Point 2: Selecting a model using existing assets

This chapter guides you through the second entry point: selecting an existing model and bringing it to IBM Z for inferencing (Figure 4-1). You follow key steps in the model development lifecycle, starting with choosing a pre-built AI model from IBM or a third-party vendor. This approach simplifies AI adoption because you can format and deploy an existing model. It enables faster integration with applications and quicker access to AI-driven insights on IBM Z and LinuxONE.

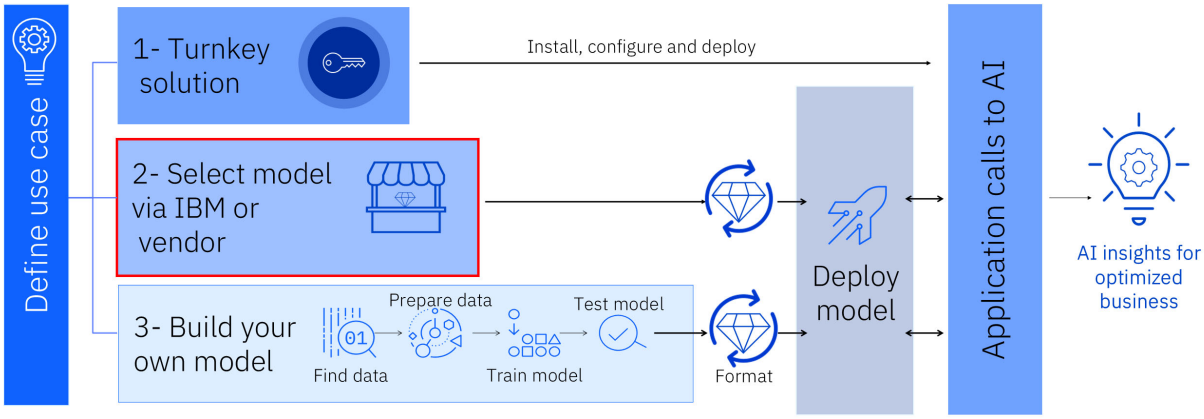


Figure 4-1 AI Entry Point 2 – Select a model via IBM or vendor

© Copyright IBM Corp. 2025.

4.1 When to choose an existing model

If you have already defined your use case and want a fast way to get started, selecting an existing model can be an effective approach. IBM provides resources to help you begin your AI journey on IBM Z with confidence. For example, if you have some data science skills and prefer to use trusted models, the second entry point—selecting a model from IBM or another vendor and deploying it on IBM Z or IBM LinuxONE—offers a practical option. This approach enables you to bring in a pre-built model, convert it as needed, and deploy it for inferencing.

You may need to convert the model to a supported format such as Predictive Model Markup Language (PMML), Open Neural Network Exchange (ONNX), or JavaScript Object Notation (JSON). You can also use the IBM Z Deep Learning Compiler with ONNX format. Once converted, you can serve the model using IBM offerings such as:

- IBM Z Accelerated Serving for TensorFlow
- IBM Z Accelerated for PyTorch
- IBM Z Accelerated for NVIDIA Triton Inference Server
- Machine Learning for IBM z/OS
- Cloud Pak for Data on IBM Z and LinuxONE

This lift-and-shift approach enables you to quickly deploy and serve models on IBM Z, accelerating your AI adoption. Figure 4-2 shows how organizations can easily select and use pre-built models from IBM or a vendor to proceed with their AI use case.

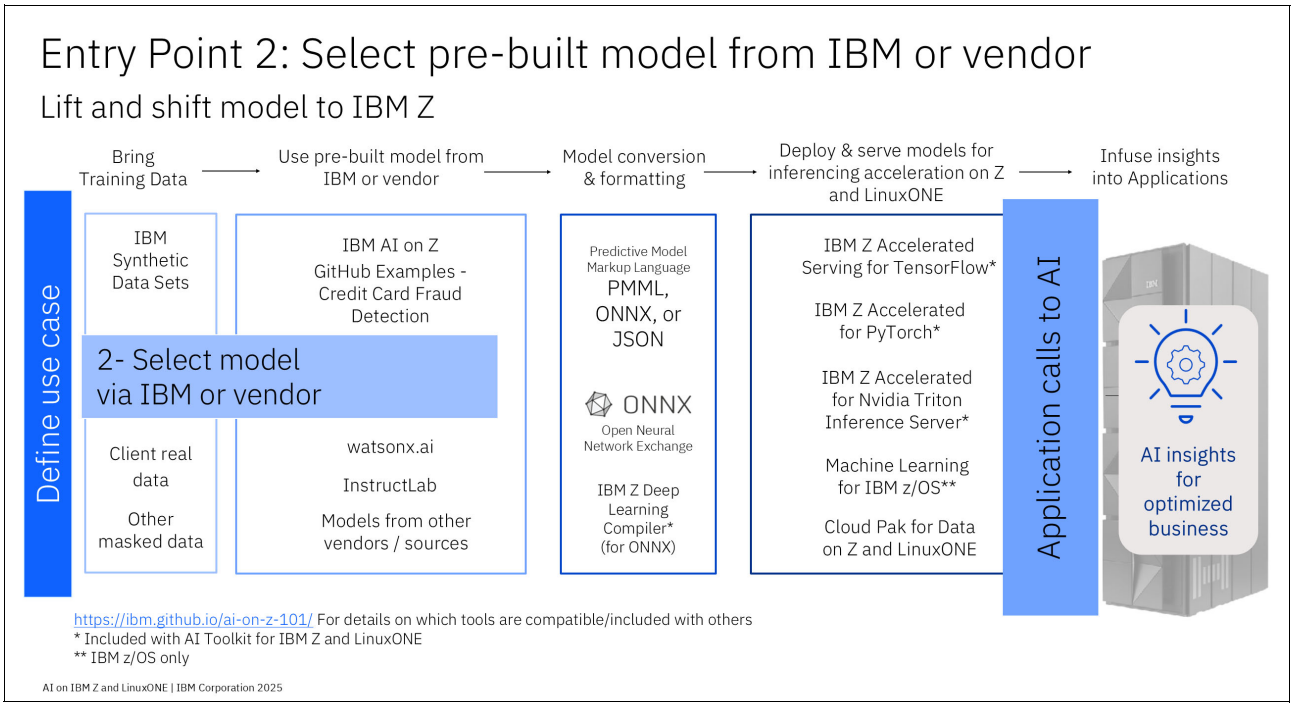


Figure 4-2 Liftand shift model to IBM Z

4.2 AI Solution Templates - AI on IBM Z GitHub examples

AI Solution Templates provide pre-built sample models and guided instructions to help you accelerate proof of concept (PoC) and prototyping for common AI use cases. These

templates include sample datasets and step-by-step guidance for deploying AI on IBM Z, enabling you to quickly validate scenarios without extensive manual setup. You can tailor the base templates to meet specific business requirements.

One example is the Credit Card Fraud Detection on IBM Z Solution Template, which demonstrates how to deploy AI in an IBM Z environment using open-source frameworks and IBM offerings such as Machine Learning for z/OS (MLz). The template begins by guiding you through building a fraud detection model using a Jupyter notebook for rapid development. You point the notebook to the dataset and run the pipeline to generate the AI model.

The solution aligns with key phases of the AI lifecycle introduced in Chapter 2, “Entry points to AI on IBM Z” on page 25, focusing on:

- Model training: Build and refine the AI model using provided datasets and frameworks.
- Model deployment: Prepare and deploy the trained model on IBM Z for inferencing.
- Model evaluation and analysis: Assess model performance and accuracy after deployment.
- Integration: Connect the deployed model with applications and workflows to deliver business insights.

After training completes, you can deploy the model with MLz or another supported platform. Deploying the model directly on IBM Z brings AI close to mission-critical workloads, which is especially beneficial for real-time scenarios. Once deployed, you can integrate the model with applications to provide inferencing APIs or enable advanced analytics. Figure 4-3 shows the Rapid AI on IBM Z Model Development Training Jupyter Notebook, available on GitHub.

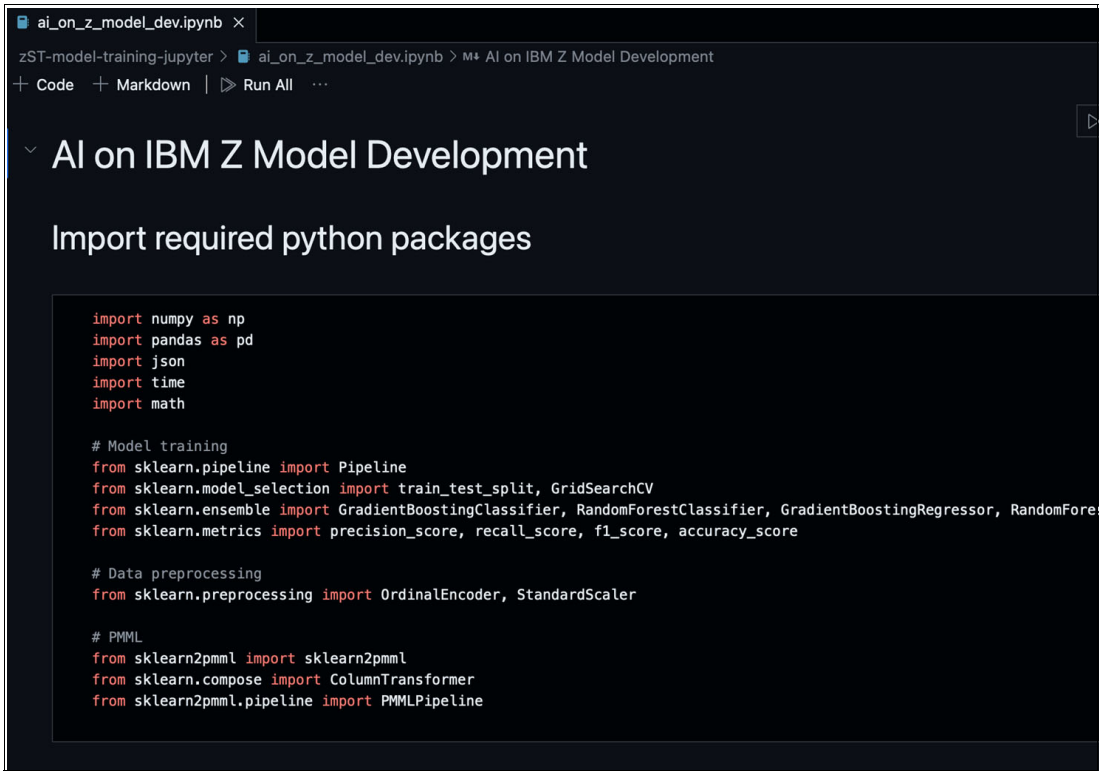


Figure 4-3 IBM Z model development training Jupyter Notebook

You need to run the entire Jupyter notebook to create the AI model. Running all the code may take some time. After the model training pipeline completes successfully, you can view the

generated AI models. The pipeline creates two models: a random forest model and a gradient boosting model.

The trained AI model can then deploy with Machine Learning for z/OS (MLz) or your deployment platform of choice. MLz is an AI model management and inferencing platform that runs natively on z/OS which is further discussed in Chapter 2 and Chapter 5. Deploying AI model directly within z/OS allows us to bring AI close to our mission critical workloads and can be beneficial especially in real-time scenarios.

Once the AI model is deployed within MLz, we can start deploying applications that use our AI model for inferencing. The application can show the latest predictions that the AI model has made. Additionally, you can use the deployed MLz fraud detection AI model and integrate it with different types of applications for further model analysis or to provide inferencing API calls.

A simple step by step instructions as the AI Solution Templates provides a simple guide to easily get started and try out the various capability on a hosted infrastructure.

Note: For the AI Solution Templates, see:
https://ambitus.github.io/aionz-solution-templates/

Or contact the AI on IBM Z team: aionz@us.ibm.com

4.3 IBM Synthetic Data Sets

IBM Synthetic Data Sets are enterprise-grade data sets that are artificially generated and do not require real seed data from clients. They are designed for AI model training and solutions that benefit IBM Z and LinuxONE clients. You can download the data sets as .CSV files and load them onto a database if needed.

Because this is a simple downloadable file, it is platform agnostic. You can use the dataset to train or validate AI models on your laptop, in a distributed environment, on IBM Z, or on IBM LinuxONE. This approach aligns with IBM's AI on Z strategy: train models anywhere and deploy them on IBM Z. That means you can use your current AI investments for training and deploy the models back to Z with Machine Learning for z/OS, AI Toolkit, or back to Linux on Z with AI Toolkit or Cloud Pak for Data on Z.

Because these are synthetic data sets, you do not need to use any real client data, which speeds up the process and helps you comply with data privacy regulations. Additionally, the data sets do not contain any real Personal Identifiable Information (PII).

In this initial release, you can access three datasets:

1. Payment Cards for Credit Card Fraud Detection
2. Core Banking & Money Laundering for Anti Money Laundering
3. Homeowners Insurance for Insurance Claims Fraud

You can use synthetic data in three ways during the AI model lifecycle:

- Building AI Models

  If you do not have a model or data and are starting from scratch, you can use synthetic data as training data to build a model. This is especially useful for Independent Service Vendors (ISVs) who do not have access to their customers' data. IBM Synthetic Data Sets

provide quick access to data, especially because you can easily use a CSV file that excludes any PII. This can lead to faster time to value.

- Enhancing AI models

  If you already have a model, you can enhance it with both synthetic and real data. Synthetic data adds richer and more diverse information that improves model quality.

- Validating AI models

  If you have a fraud model and want to test its prediction accuracy, you can use synthetic data sets as an answer key. These datasets label all transactions for fraud, so the ground truth is known.

For more information on Synthetic Data Sets, see:

- “IBM Synthetic Data Sets” on page 55
- *IBM Synthetic Data Sets*, REDP-5748
- IBM Synthetic Data Sets

4.4 Large Language Models support with IBM Telum II and Spyre

The IBM z17, IBM Telum® II, and Spyre AI Accelerator power new AI use cases that transform and simplify your experience. With the IBM Telum II Processor and the Spyre AI Accelerator, predictive and generative AI and Large Language Models (LLMs) provide multi-model AI for real-time, high-volume transaction processing with greater precision and accuracy. These capabilities help you solve advanced use cases such as enhanced fraud detection, anti-money laundering, and anomaly detection.

These advanced AI capabilities combine traditional AI models with encoder LLMs, which are non-generative. Encoder LLMs are not trained to predict; instead, they help you find and extract useful information from unstructured data such as text and documents. The IBM Telum II on-chip accelerator supports encoder LLMs, enabling you to apply advanced predictive AI use cases. You can also use the Spyre card to support encoder LLMs that are complex or operate under stringent SLAs.

Figure 4-4 shows Predictive AI, Multiple Model, and Generative AI use cases with the IBM Telum II best supporting Predictive AI and Multi Model use cases and the Spyre AI Accelerator best supporting Generative AI use cases.

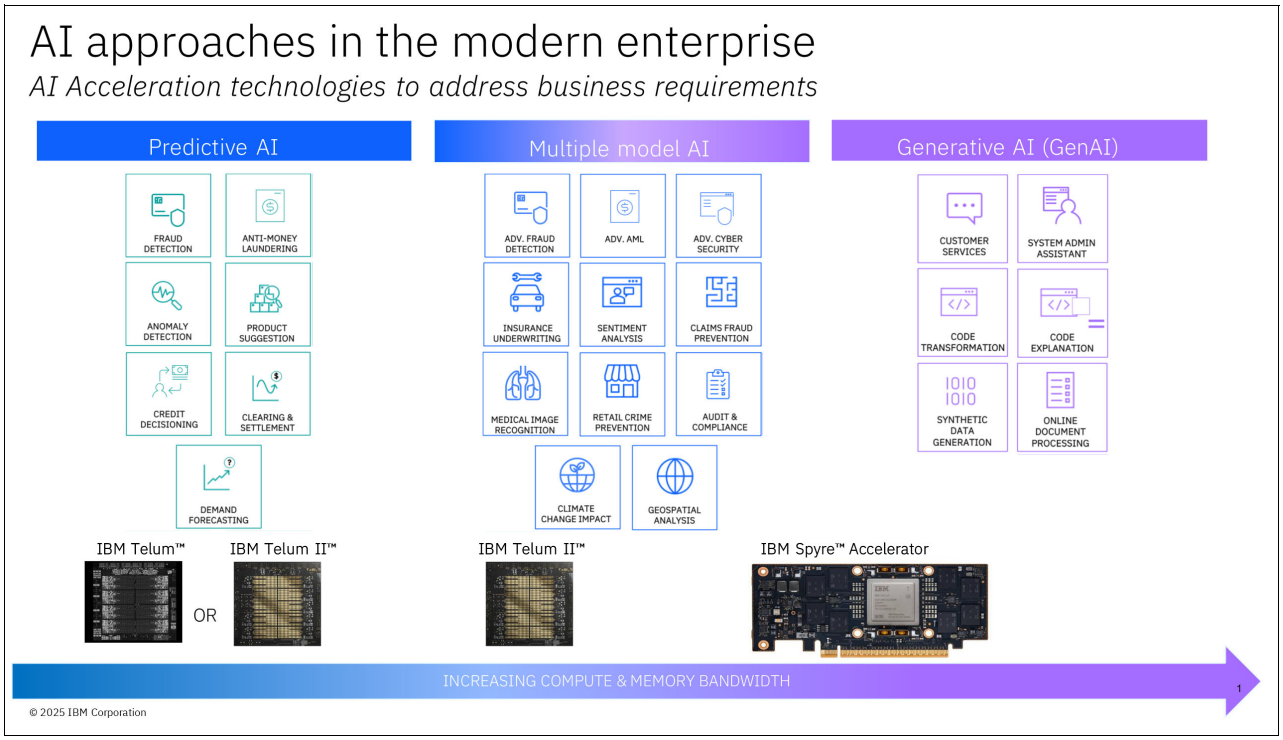
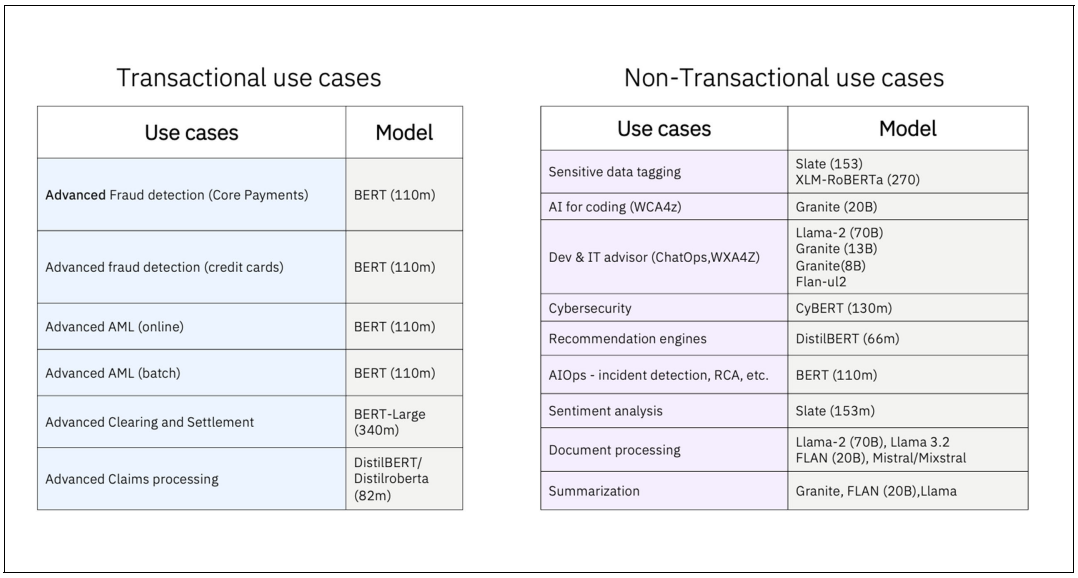


Figure 4-4 AI use cases across predictive, multi-model, and generative AI enabled by IBM Telum II and Spyre accelerators.

Additionally, Decoder LLMs are usually used to generate content, responses, and text and to support the implementation of chatbots and assistants. This type of LLMs can accelerate and scale Generative AI workloads in a secure, optimized environment to solve use cases such as improving productivity and skills onboarding or even accelerating application modernization. With the Spyre technology offers a level of AI acceleration that designed to support decoder Large language models. This level of AI acceleration begins to unlock new AI possibilities on the platform. We can now, not only use AI inferencing to score every transaction but we can fold in Large Language Models to that process for added insights.

Figure 4-5 shows different transactional and non-transactional use cases and the different Large Language Models that can be implemented for each use case.



Transactional use cases

Use cases	Model
Advanced Fraud detection (Core Payments)	BERT (110m)
Advanced fraud detection (credit cards)	BERT (110m)
Advanced AML (online)	BERT (110m)
Advanced AML (batch)	BERT (110m)
Advanced Clearing and Settlement	BERT-Large (340m)
Advanced Claims processing	DistilBERT/ Distilroberta (82m)

Non-Transactional use cases

Use cases	Model
Sensitive data tagging	Slate (153) XLM-RoBERTa (270)
AI for coding (WCA4z)	Granite (20B)
Dev & IT advisor (ChatOps,WXA4Z)	Llama-2 (70B) Granite (13B) Granite(8B) Flan-ul2
Cybersecurity	CyBERT (130m)
Recommendation engines	DistilBERT (66m)
AIOps - incident detection, RCA, etc.	BERT (110m)
Sentiment analysis	Slate (153m)
Document processing	Llama-2 (70B), Llama 3.2 FLAN (20B), Mistral/Mixstral
Summarization	Granite, FLAN (20B),Llama

Figure 4-5 Transactional vs. non-transactional AI use cases and models.

4.5 Getting started

There are three easy ways for you to get started:

1. Client Briefing – Discuss and see a demonstration of AI on IBM Z strategy and capabilities. Learn how you can leverage AI to create impact in your business (1–2 hours, onsite or virtual).
2. Discovery Workshop – Engage with the AI on IBM Z expert team to identify and ideate on candidate use cases. Define the value and scope of a project architecture (½–1 day, onsite or virtual).
3. Pilot / Proof of Concept (PoC) – Collaborate with a multi-disciplinary IBM team to innovate and rapidly demonstrate AI on IBM Z (4–6 weeks).

5

Entry Point 3: Build your own - starting from scratch

Building your own AI model from scratch gives you the flexibility to tailor solutions to specific business challenges although requiring data science expertise and tools. This approach is particularly valuable when you require full control over the model creation and training process . IBM Z supports this strategy by enabling you to train the model on any platform inclduing IBM Z and operationalizing it afterwards on IBM Z, co-located with transactional applications and data, thereby enhancing performance and reducing latency.

In this chapter we discuss the third entry point, building your own model (Figure 5-1). This chapter guides you through the key steps in the full model development lifecycle: finding data, preparing it, training. and testing. This gives you flexibility to build custom AI models using familiar tools and frameworks that are also available on IBM Z and LinuxONE. These steps are part of a broader process. You define the use case, install and configure the solution, deploy the model, and integrate AI into applications to generate business insights.

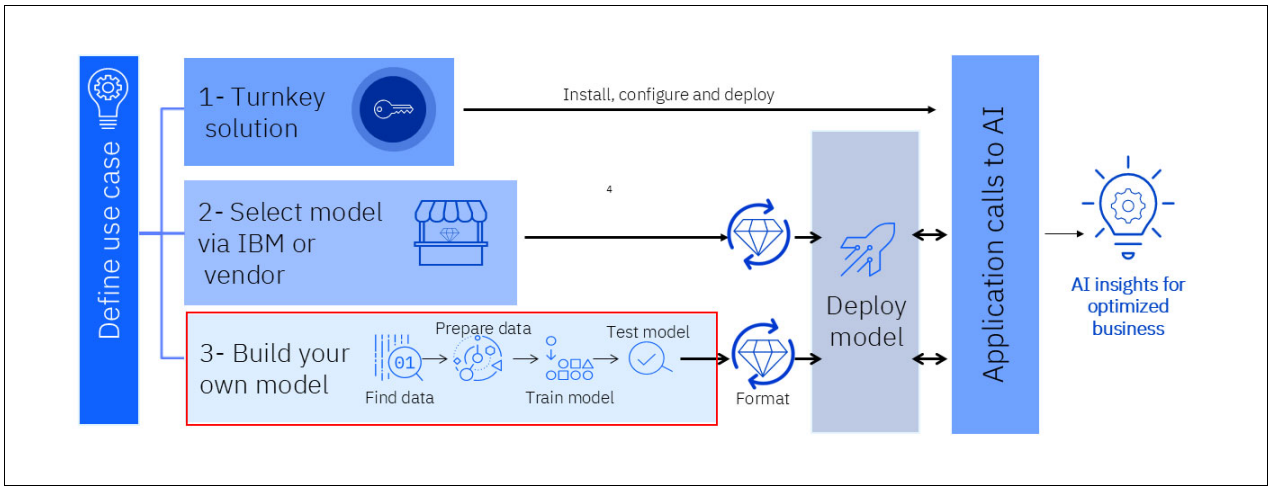


Figure 5-1 AI Entry Point 3 – Build Your Own Model

© Copyright IBM Corp. 2025.

5.1 Build your own model - starting from scratch

Your AI journey starts when you identify a meaningful use case that helps you address a specific business challenge. Once the use case has been defined, the organization can opt to build the target AI model from scratch leveraging their tools of choice and platform of choice, including IBM Z. This makes especially sense when businesses already have AI expertise and data science skills in house and/ or if they need a higher control over model creation and training process.

The AI on IBM Z strategy is intended to support clients to train and build AI models anywhere and deploy them on IBM Z to benefit from the co-location of AI processing with transactional applications and data.

If clients opt to build their models on IBM Z, they can select one of the supported popular frameworks and open-source technologies that their data scientists and AI experts are familiar with. The supported tools on IBM Z offer the same user experience as on other platforms without requiring additional platform skills.

The investment in the tools and the expertise in house can be leveraged in the future to implement and build additional AI use cases which leads to higher return of investment.

Figure 5-2 shows the different technologies and offerings that support each phase of the AI model lifecycle from defining the use case and scoping the AI project, to bringing training data, building AI models, converting the model to a suitable format before deploying and serving it for inference, until infusing the AI processing into the target application.

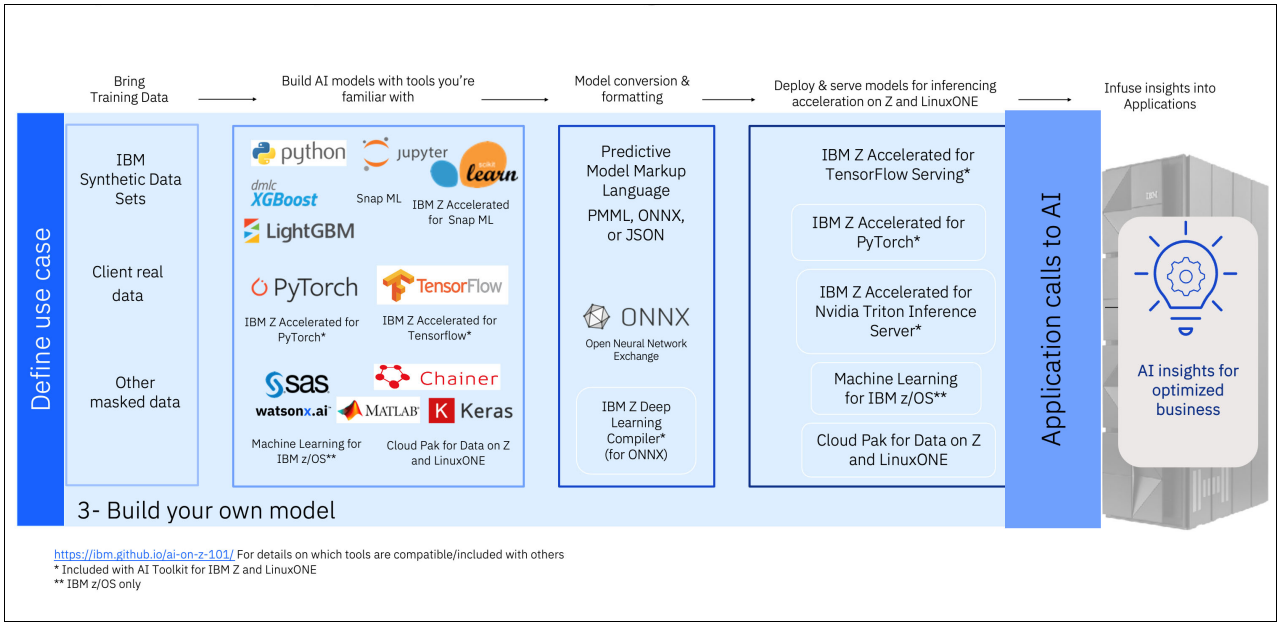


Figure 5-2 Entry Point 3 - Build your own model, starting from scratch

Note: For additional information, see Journey to AI on IBM Z and LinuxONE .

5.1.1 Data selection for AI model training

Based on the defined use case, you need to determine which data is required for implementation and model training. During this exploration phase, it's also important to define how key stakeholders will access the data needed for use case implementation and deployment. You can choose to use your own real data, leverage IBM Synthetic Datasets, or work with other masked data sources.

Real data

Using real data that is collected from real-world sources for training brings advantages like identifying patterns and correlations between variables that can help to gain accurate insights which reflect real-world scenarios. However, identifying where the data resides and how to get access to it might be challenging. In several organizations, data scientists and AI experts struggle to get access to their real data due to specific regulations and limitations.

In addition, real data can contain sensitive and identifiable information. It may also include biases that reflect actual populations and data collection methods, increasing the risk of producing biased models. To address data sensitivity, you may need to invest additional effort in cleansing the data and applying data masking techniques.

To help you access real data stored on IBM Z, IBM provides several technologies that make data accessible, consumable, and ready for processing in platforms like watsonx.ai or Cloud Pak for Data. The Data Gate family simplifies data synchronization and access. Additionally, Data Virtualization Manager (DVM) offers a convenient way to access data across hybrid cloud, analytics, and AI environments. By leveraging in-memory data virtualization, DVM supports various data sources, including Db2 for z/OS, IMS, VSAM, Oracle, Adabas, and non–IBM Z sources, helping you accelerate application development and reduce the cost of modernizing data access.

Note: For more details, see the following:

- *IBM Data Virtualization Manager for z/OS*, SG24-8514
- IBM Documentation

IBM Synthetic Data Sets

Synthetic data can serve as training data to build models from scratch. Using synthetic data allows you to overcome regulatory restrictions on real data because it protects privacy and prevents real data from leaking. In addition, quick data access enables shorter implementation and delivery cycles.

IBM Synthetic Data Sets includes enterprise-grade data sets that you can use for AI model training, large language model (LLM) fine-tuning, and enhancing existing models for solutions that benefit IBM Z and IBM LinuxONE clients. This offering helps you accelerate and improve AI model development with artificial data tailored for the financial services industry to reduce time to value and avoid risk.

IBM Synthetic Data Sets includes payment card data that you can use to build models for credit card fraud detection. The offering also provides core banking and money laundering data to train anti-money laundering (AML) models, as well as homeowners insurance data that you can use to implement insurance claims fraud models.

In addition to using IBM Synthetic Data Sets, you can work with other masked data to overcome challenges related to accessing or using real data.

Before you build the model, explore tools and techniques that help you achieve the desired results. If your use case requires predictive AI insights, identify the type of task the model should perform—such as classification, regression, clustering, or anomaly detection. You also need to select appropriate techniques to explore the data, detect patterns, and apply feature engineering if you plan to use traditional machine learning models. These steps help you prepare and fit the data for use in the model and represent a critical phase of model planning. After you complete these steps, you can build the model.

5.1.2 Model building

The model building phase involves creating a model using historical data and selecting an algorithm that delivers accurate and relevant business insights. You follow an iterative process of training, testing, and comparing results to refine the model.

Figure 5-3 illustrates the key phases of the machine learning and AI model lifecycle. It highlights the model building steps, which typically involve using prepared data—following data exploration and pre-processing—to train, test, and validate the model.

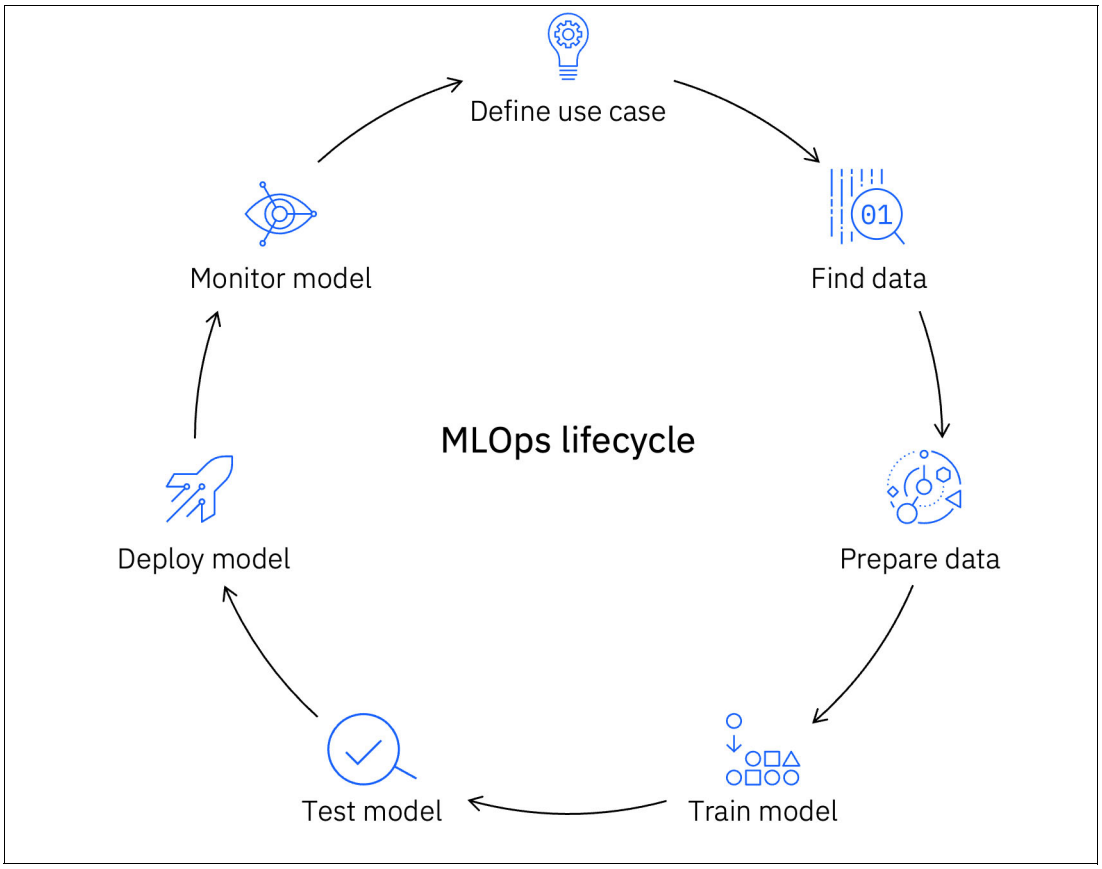


Figure 5-3 Machine learning operations (MLOps) lifecycle

Data scientists and AI engineers can use their tools of choice to build their AI models. This applies especially to AI model types like machine learning and deep learning. If the use case requires a large language model (LLM), you can leverage pre-trained LLMs and fine-tune them for specific tasks. Building an LLM from scratch is not recommended due to the high cost and complexity involved in training.

To build a machine learning or deep learning model, you can choose from popular open-source frameworks such as PyTorch, TensorFlow, Scikit-learn, Keras, or LightGBM. You can also leverage your existing tool investments, including vendor solutions like MATLAB and SAS, or IBM offerings such as Machine Learning for IBM z/OS and Cloud Pak for Data on IBM Z and LinuxONE.

Model building on IBM Z

IBM Z supports a rich set of tools for building AI models directly on the mainframe. If you choose to develop models in this environment, you can take advantage of several leading open-source technologies available on the IBM Z platform. The following examples highlight some of these tools, though the list is not exhaustive.

Note: You can view the product overview in Figure 2-3 in section 2.2, “AI on IBM Z products used in the AI lifecycle” on page 27 for a list of some of the outlined products.

- PyTorch

  PyTorch is widely used for deep learning research and language processing. It is a fully featured framework for building deep learning models and supports large language models (LLMs). PyTorch capabilities are included in the AI Toolkit for IBM Z and LinuxONE. You can run it as a container image on Linux on IBM Z or within IBM Z Container Extensions.

- TensorFlow

  TensorFlow offers open-source deep learning tools and frameworks for data scientists. It supports a wide range of use cases, including natural language processing (NLP), image recognition, voice search, and more. TensorFlow is included in the AI Toolkit for IBM Z and LinuxONE and is delivered through the IBM Z Accelerated for TensorFlow component.

- SnapML

  To build linear, tree-based, or gradient boosting models, you can use SnapML—an open-source library developed and maintained by IBM Research®. SnapML supports training and inference for popular machine learning models and is included in the AI Toolkit for IBM Z and LinuxONE. It is also part of the IBM Z Accelerated SnapML offering.

- Apache Spark

  Apache Spark, available in the IBM Z Platform for Apache Spark offering, includes a Machine Learning Library (MLib) that trains and builds traditional AI models in Scala, Python, or Java. The corresponding data preprocessing workloads are eligible for IBM z Integrated Information Processor (zIIP) offload, with varying percentages depending on the programming language..

- Machine Learning for IBM z/OS

  Machine Learning for IBM z/OS is an enterprise offering that provides an end-to-end platform supporting the full AI lifecycle and running natively on z/OS. It allows you to deploy models built anywhere, including on IBM Z.

  If you build AI models on Machine Learning for IBM z/OS, you benefit from a multi-user model development environment that uses JupyterHub. This framework provides a dedicated workspace for data scientists, supports both Spark and Python, enables quick access to z/OS data, and lets you easily save the trained model into Machine Learning for IBM z/OS. In addition, the training workload is highly eligible for ZIIP offload.

- IBM Watson® Studio within Cloud Pak for Data

Cloud Pak for Data is an established cloud-native data and AI platform that helps modernize data management, analytics, and AI. IBM Watson Studio, a component of Cloud Pak for Data, enables you to build AI models.

It includes an open, productive integrated development environment (Jupyter Notebooks, JupyterLab) and automation tools (such as AutoAI) in a unified workspace. It provides dozens of connectors and multiple deployment options to help you connect to and analyze data quickly.

Model building on other platforms

You can benefit from the AI agility concept for frictionless AI across platforms. This approach enables you to train and build AI models on your preferred platform and deploy them on IBM Z for accelerated inferencing.

This AI agility aligns with many organizations' hybrid cloud infrastructure, especially when model building and training already occur on other platforms. In such cases, you do not need to change your existing model-building processes. However, deploying and inferring on IBM Z is recommended to gain the advantages discussed earlier.

Note: Db2 Data Gate and Data Gate for watsonx are two solutions enabling trusted read-only synchronization of IBM z/OS data sources to hybrid cloud infrastructures. For more information, see https://www.ibm.com/products/data-gate .

5.1.3 AI model formatting

After training and building the model, data scientists usually convert it to a standard format. This step ensures that the model can be efficiently ported and deployed across different AI platforms and systems, enabling flexibility and interoperability across your organization's environments.

Standard formats such as PMML, ONNX, and JSON allow you to share, port, and deploy models across various hardware systems, including IBM Z. On IBM Z, you can optimize deployment to leverage integrated AI acceleration on Telum and Telum II processors..

Note: Various frameworks for model conversion are available, like tf2onnx. tf2onnx is a framework to convert a TensorFlow model to the ONNX format. For more information, see https://github.com/onnx/tensorflow-onnx.

5.2 Use case example 1: Banking

This section presents a real-world example of applying AI for fraud detection in instant payments, demonstrating how co-locating AI inference with transactional data on IBM Z enables real-time decision-making and operational efficiency.

Business requirements

A banking IT organization needs to address the requirement of identifying transaction fraud risk within instant payments allowing them to make informed decisions of accepting or declining transactions at the point of impact. This requires applying in-transaction AI for fraud detection at scale which means that they must scan every transaction that takes place on their IBM Z environment. They needed an approach that scans transactions in real time and improves service quality and operational efficiency by reducing manual processing.

Their existing approach uses AI models to prioritize potentially fraudulent payment transactions for later post-processing. These AI models run on IBM Cloud® Pak for Data and are trained for fraud detection. To meet instant payment fraud detection requirements, the AI models must not only prioritize but also detect and handle fraudulent payment transactions in real time to prevent fraud risk.

Solution approach

Applying fraud detection alongside their transactional application and data addresses real-time and service quality requirements. The solution infuses AI inference into the IBM Z transaction runtime by using IBM Machine Learning for IBM z/OS, an end-to-end enterprise AI platform for training, building, deploying, and monitoring AI models on IBM Z.

Model building and deployment

The team responsible for building the model uses historical data and IBM Cloud Pak® for Data to build a tree-based model. They can port and deploy the model on Machine Learning for IBM z/OS after converting it to Predictive Model Markup Language (PMML) format. This approach allows them to preserve existing investments and processes to build and re-train the model on their preferred platform their preferred tools. In addition, deploying the AI model on IBM Z lets the inference process use integrated Telum AI acceleration technology.

Furthermore, you can use WebSphere Optimized Local Adapter (WOLA) technology, which lets you perform fast, efficient, low-latency memory-to-memory exchanges between Java applications in WAS/Liberty z/OS and programs running in external address spaces such as CICS, IMS, and Batch. WOLA is an excellent communication method between transactional and AI workloads that require high throughput. Calls to other platforms were tested but did not meet the required response time.

Both AI acceleration and the WOLA mechanism help deliver fraud detection insights at speed and scale.

5.3 Use case example 2: Insurance

This section shows how you can apply AI in the insurance industry to streamline claims processing and reduce fraud risk. It demonstrates how combining structured data with insights extracted from unstructured text using a fine-tuned large language model enables automated adjudication and faster decision-making on IBM Z.

Business requirements

The second use case involves a large home insurance provider that improves claims processing and optimizes the claim lifecycle while reducing fraud risk. This optimization helps you retain customers, increase satisfaction, and handle workload peaks caused by natural catastrophes or special events. High workloads can lead to longer waiting times because manual processing is required for both unstructured and structured data.

You can already predict claim risk based on structured policy data, but this information might miss important insights often included in the claimant’s text description. These insights require additional human processing to understand key aspects such as the cause or urgency of the claim.

Solution approach

The solution automatically adjudicates low-risk claims, which lets you reduce adjustment time and manual investigation work. To optimize and accelerate claims processing, you can use an

encoder-based large language model to read claim text and extract key insights. These insights are categorized or ranked to provide information such as the cause and urgency of the claim, and then appended to the structured feature table based on customer profile data. A predictive AI model then assesses the claim risk. Based on this approach, you can automatically adjudicate low-risk claims, reducing manual effort and optimizing processing time.

Model building and deployment

Figure 5-4 shows how you can streamline insurance claims processing by using key technologies from the AI Toolkit for IBM Z and LinuxONE. It demonstrates how structured and unstructured data sources work together to build, serve, and integrate AI models into a claims processing application.

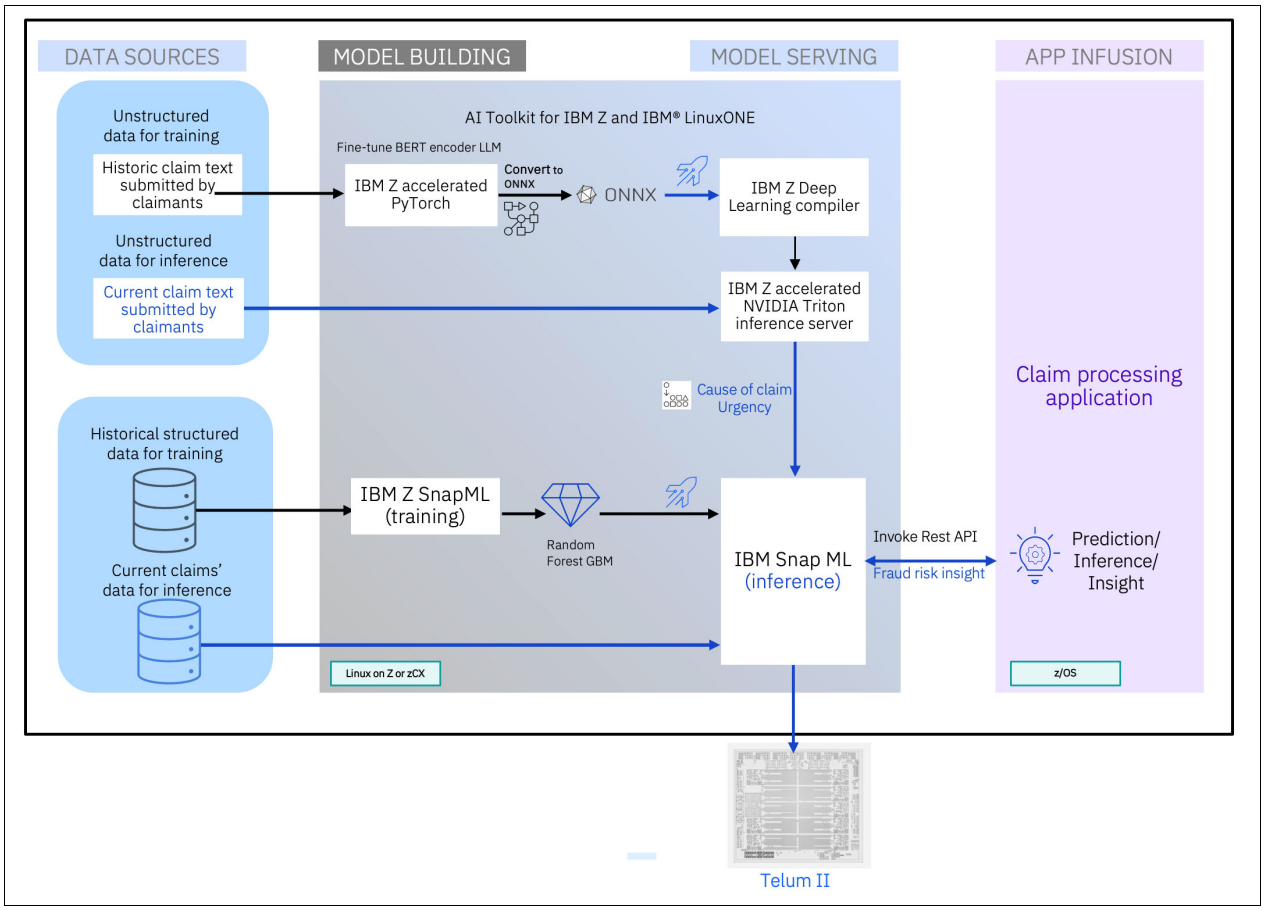


Figure 5-4 Insurance claim processing - AI model lifecycle

To optimize insurance claims processing and reduce fraud risk, you combine structured policy data with insights extracted from unstructured claimant text. This approach enables automated adjudication of low-risk claims and accelerates decision-making.

You train two components in parallel to achieve this goal:

- Predictive AI model: Use IBM Z Accelerated SnapML to train and deploy classification models—such as random forest and gradient boosting—on cleansed, structured claims data. These models forecast claims fraud risk and operate efficiently on IBM Z for real-time scoring.

- Encoder-based LLM: Fine-tune a Bidirectional Encoder Representations from Transformers (BERT) model on historical claimant text using IBM Z Accelerated PyTorch, available in the AI Toolkit for IBM Z and LinuxONE. BERT is widely used for non-generative tasks such as extracting information from long documents. It identifies context-driven signals like cause of damage and urgency, which you append to the structured feature set. After fine-tuning, convert the model to the ONNX format, compile it with the IBM Z Deep Learning Compiler, and deploy it using the IBM Z Accelerated NVIDIA Triton Inference Server optimized for Telum II AI acceleration.

By enriching structured data with insights from the encoder LLM, you enable the SnapML predictive model to score claims more accurately, identify high-risk patterns, and flag potential fraud. This integrated approach combines advanced AI techniques with IBM Z's performance and security, delivering faster, more reliable claims processing.This use case shows how you can combine entry points 2 and 3 in a multiple-model implementation.

5.4 Other resources to get started with AI on IBM Z

To help you accelerate time to value and support your AI journey on IBM Z, IBM provides resources that guide you in identifying meaningful use cases and starting AI implementation in your IBM Z environment. This section outlines some of these resources.

Note: You can find general information about AI on IBM Z and additional resources outside this publication at:

IBM AI on Z and AI on Z 101

5.4.1 Getting started with use case identification - AI on IBM Z Discovery workshop

Because your AI journey typically begins with identifying a meaningful use case, IBM offers to facilitate and guide ideation through the AI Discovery Workshop. This no-cost offering helps you explore and evaluate potential AI use cases. Conducted over one to two days, the workshop provides a comprehensive overview of the business value and technologies required for AI on IBM Z. It helps you identify a feasible use case and define the scope for a proof of concept (POC) to support your initial implementation. This approach gives you a practical starting point for adopting AI on IBM Z.

AI projects require collaboration across multiple skill sets, including AI expertise, domain knowledge, infrastructure deployment, and governance. Without the right mix of expertise, defining a meaningful AI use case can be challenging. The AI Discovery Workshop depends on collaboration among individuals who understand business logic, data, AI, and infrastructure. To ensure productive outcomes, you should involve a diverse group of participants, such as application engineers, site reliability or infrastructure engineers, line-of-business representatives, data architects, and data scientists.

The IBM team provides a preliminary agenda that you can fully customize based on your organization's specific needs and objectives, ensuring the workshop delivers maximum value. The workshop demonstrates the business value of performing analytical scoring and inferencing directly on IBM Z. It guides you through the process of taking a model trained

using industry-standard analytical frameworks off-platform and deploying it seamlessly onto IBM Z.

Note: To request a discovery workshop, contact your IBM representative or contact aionz@us.ibm.com.

Support during the implementation - AI solution templates

AI Solution Templates on IBM Z and LinuxONE are pre-built blueprints that help you accelerate time to value by jumpstarting AI deployment through rapid prototyping. These templates guide you through the full AI lifecycle on IBM Z and LinuxONE across various enterprise use cases, using a range of technologies. Whether you are a senior data scientist or new to AI, you can build an AI model, deploy it on IBM Z, and integrate it into a business application.

This is made possible through a guided experience for each use case, including sample synthetic datasets, AI model training pipelines, sample application code, and more. These templates enable you to deploy AI models efficiently without requiring significant investment in data science expertise or manual feature engineering. You can deploy AI models on IBM Z and take advantage of the IBM Z Integrated Accelerator for AI.

AI Solution Templates represent the culmination of extensive knowledge, domain expertise, and experience gained through years of AI implementation.

AI Solution Templates on IBM Z and LinuxONE are available through the Open Mainframe Project. These templates include implementations of popular use cases on the platform, such as fraud detection using Machine Learning for z/OS, credit risk assessment, health insurance claims processing, and fraud detection using the AI Toolkit.

Note: For more details, see https://ibm.github.io/ai-on-z-101/solutiontemplates/ for AI Solution Templates on IBM Z and LinuxONE.

Related publications

The publications listed in this section are considered particularly suitable for a more detailed discussion of the topics covered in this paper.

IBM Redbooks

The following IBM Redbooks publications provide additional information about the topic in this document. Note that some publications referenced in this list might be available in softcopy only.

- *AI for Linux on IBM Z and LinuxONE Applications and Examples*, REDP-5756

  https://www.redbooks.ibm.com/abstracts/redp5756.html
- *Securely Leverage Open-Source Software with Python AI Toolkit for IBM z/OS*, REDP-5709

  https://www.redbooks.ibm.com/abstracts/redp5709.html
- IBM z17 (9175) Technical Guide, SG24-8579

  https://www.redbooks.ibm.com/abstracts/sg248579.html
- IBM Db2 13 for z/OS and More, SG24-8527

  https://www.ibm.com/docs/en/db2-for-zos/13.0.0?topic=running-ai-queries-sql-data-insights
- IBM Data Virtualization Manager for z/OS, SG24-8514

  https://www.ibm.com/docs/en/dvm/1.2.0
- IBM Redbooks: IBM z16 Technical Overview: https://www.redbooks.ibm.com/abstracts/redp5749.html
- IBM Redbooks: IBM z16 System Recovery Boost: https://www.redbooks.ibm.com/abstracts/redp5661.html
- IBM Redbooks: IBM z16 System Overview: https://www.redbooks.ibm.com/abstracts/sg248552.html

You can search for, view, download or order these documents and other Redbooks, Redpapers, Web Docs, draft and additional materials, at the following website:

ibm.com/redbooks

Other publications

These publications are also relevant as further information sources:

- *IBM Synthetic Data Sets*, REDP-5748

  https://www.redbooks.ibm.com/redpapers/pdfs/redp5748.pdf

© Copyright IBM Corp. 2025.

Online resources

These websites are also relevant as further information sources:

- AI on IBM Z Overview: https://www.ibm.com/products/z/artificial-intelligence
- IBM Z Feature Page (z17): https://www.redbooks.ibm.com/feature/z17
- AI Solution Templates: https://ambitus.github.io/aionz-solution-templates/
- AI on Z 101: https://ibm.github.io/ai-on-z-101/solutiontemplates/
- Python AI Toolkit for IBM z/OS: https://ibm-z-oss-oda.github.io/python_ai_toolkit_zos/
- IBM Container Registry for AI Toolkit: https://www.ibm.com/products/ai-toolkit-for-z-and-linuxone
- Fraud Detection Overview: https://www.ibm.com/think/topics/fraud-detection
- AI Fraud Detection in Banking: https://www.ibm.com/think/topics/ai-fraud-detection-in-banking
- IBM Spyre Accelerator Info: https://www.ibm.com/support/z-content-solutions/spyre-accelerator-z-and-linuxone/
- tf2onnx Model Conversion: https://github.com/onnx/tensorflow-onnx
-

Help from IBM

IBM Support and downloads

ibm.com/support

IBM Global Services

ibm.com/services



REDP-5758-00

ISBN

Printed in U.S.A.

Get connected



