Draft Document for Review May 22, 2025 2:51 pm REDP-5756-00

Redbooks
ibm.com/redbooks

AI for Linux on IBM Z and LinuxONE Applications and Examples

Lydia Parziale

Jasmeet Bhatia

Shrirang Kulkarni

Anna Shugol

Abraham Varghese

Markus Wolff

Kelly Yang

IBM Z

IBM LinuxONE

Artificial Intelligence

IBM

Redpaper

IBM Redbooks

AI for Linux on IBM Z and LinuxONE

May 2025

REDP-5756-00

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

First Edition (May 2025)

This edition applies to LinuxONE 4 and LinuxONE 5, IBM z/16 and z17 Telum 1 and Telum 2.

This document was created or updated on May 22, 2025.

Contents

Notices

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.

Code Assistant™
Db2®
DB2®
IBM®
IBM Cloud®
IBM Cloud Pak®
IBM Research®
IBM Spyre™
IBM Telum®
IBM Watson®
IBM watsonx®
IBM Z®
IBM z16®
IBM z17™
Instana®
Redbooks®
Redbooks (logo) ®
Spyre™
watsonx®
watsonx Assistant™
watsonx Code Assistant™
watsonx.data®
watsonx.governance®
z/OS®
z/VM®
z15®
z16®
z17™
zSystems™

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, RHCE, 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

Ever wonder how AI could turbocharge your enterprise without compromising on security or flexibility? Explore the realm of AI for Linux on IBM Z and LinuxONE and jump start your AI projects by exploring the potential of using LinuxONE as a robust platform for developing and deploying AI applications.

This IBM Redbooks publication is a practical guide that will provide real examples and use cases to demonstrate how to leverage the efficiency and security of Linux on IBM Z and LinuxONE to accelerate your AI journey and AI initiatives. It will also assist readers in gaining an understanding of where the entry points are and what steps are involved in deploying an AI application.

This IBM Redbooks publication gives you a front-row seat to how AI can solve big problems and drive innovation on some of the most powerful hardware out there and includes the following topics:

- Chapter 1, "Introduction: The convergence of AI and enterprise Linux systems" on page 1
- Chapter 2, "Optimized Model Serving Solutions" on page 9
- Chapter 3, "Building enterprise ready AI applications at scale" on page 23
- Chapter 4, "Bringing it all together with the use cases" on page 37

Whether you're a developer, a business owner, or just curious about the future, this IBM Redbooks publication provides the information on how implementing AI in your enterprise can solve big problems and drive innovation on some of the most powerful hardware out there.

Throughout this publication, we will refer to Linux on IBM Z and LinuxONE simply as Linux, since both run on the s390x architecture

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 for the IBM® Redbooks® team in Poughkeepsie, New York, with domestic and international experience in technology management including software development, project leadership, and strategic planning. Her areas of expertise include business development and database management technologies. Lydia is a PMI certified PMP and an IBM Certified IT Specialist with an MBA in Technology Management and has been employed by IBM for over 30 years in various technology areas.

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 hands-on 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 doctoral 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.

Shrirang Kulkarni is LinuxONE and Cloud Architect at IBM with over 20 years of experience in IT architecture, specializing in Linux on System Z, IBM z/VM®, and hybrid cloud solutions. I've led enterprise transformation projects across 25+ countries, working with global clients and system integrators. As an IBM Expert Level IT Specialist, Open Group Certified Master, and Red Hat Certified Engineer (RHCE), I bring both technical depth and strategic vision. I've authored four IBM Redbooks and several articles focused on cloud, containers, and secure infrastructure. My current work includes driving Red Hat OpenShift deployments, zCX solutions, and Hyper Protect Services on IBM Z. I'm passionate about solving real-world challenges with resilient, scalable architectures.

Anna Shugol is a Senior Data and AI on IBM Z Solutions Engineer, a part of worldwide zAcceleration team. In her current role she helps clients with adoption of Data and AI solutions on IBM z/OS® and Linux on IBM Z.

Anna joined IBM mainframe technical team in 2011 and since then participated and led various complex mainframe client projects.

Anna's professional interest is designing and implementing performant, scalable, secure and sustainable AI on Z infrastructure.

Abraham Varghese is the Product Manager for AI on IBM Z and LinuxONE, with over 25 years of experience in development and transformation projects across the IBM Z and cloud ecosystem. He plays a strategic role in accelerating enterprise transformation and mainframe modernization for IBM Z and LinuxONE.

Currently a research scholar pursuing a Doctorate in Artificial Intelligence, Abraham brings both academic insight and hands-on leadership to his work. He has led numerous cloud development projects in the past and has been deeply involved in IBM Z modernization initiatives—collaborating closely with Global System Integrators, Managed Service Providers and the Global Capability Centre to modernize Mainframe application and integrate next gen capabilities.

As a recognized IBM Z Advocate, Abraham actively promotes IBM Z technologies, mentors emerging talent, and contributes to the growth of the global Z community. He has authored several technical blogs and co-authored white papers with system integrators.

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 machine learning in regard to IBM Machine Learning for z/OS, Db2® AI for z/OS, Cloud Pak for Data and the IBM watsonx® portfolio.

He has contributed to previous IBM Redpapers regarding AI on IBM Z, such as "Optimized Inferencing and Integration with AI on IBM zSystems™: Introduction, Methodology, and Use Cases" and is part of the worldwide AI on IBM Z SWAT team, lead by IBM Fellow Elpida Tzortzatos.

Kelly Yang is a Senior Product Manager for AI on IBM Z & LinuxONE in Poughkeepsie, NY, United States. She has 5 years of experience in the AI on IBM Z and LinuxONE field. She has worked at IBM for 9 years. Her areas of expertise include a B.S in Computer Science and MBA from Clarkson University.

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

Rhonda Sundlof, Product Management for AI on IBM Z and LinuxONE
IBM Poughkeepsie, USA

Calvin Friedrich, Data & AI on IBM Z, Global Sales - Infrastructure Sales
Markus Leber, Data & AI on zSystems Technical Sales Software Architect, IBM Technology Sales, DACH
Lukas Maly,zStack LinuxONE Platform Sales, IBM Technology DACH
Rudiger Stumm, Senior zData&AI Technical Specialist Manager, IBM Technology Sales, DACH
IBM Boeblingen, Germany

Abhiram Kulkarni, Software Architect, Hyper Protect
IBM Bangalore, IN

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

Introduction: The convergence of AI and enterprise Linux systems

Artificial Intelligence (AI) is a technology that refers to the simulation of human intelligence that is capable of learning from the data, recognizing and making decisions like humans and has the ability to improve over time.

AI systems can process large amounts of data that will take decisions based on trends and make predictions. AI is designed to perform tasks that typically require human intelligence, such as speech recognition, visual perception and language translation. AI includes a wide range of technologies such as natural language processing, machine learning, robotics, and computer vision.

AI works by enabling machines to mimic human intelligence and perform tasks that typically require human cognition. At its core, AI relies on a combination of data, algorithms, and computational power.

The following is a basic summary of how AI works.

- **Data gathering:** AI is fed large amounts of data from various sources such as text, numbers and images.
- **Learning:** AI uses algorithms to find patterns, relationships or rules in the data such as Machine Learning and Deep Learning.
- **Processing and decision making:** AI creates a model and based on learned patterns, AI makes predictions or suggestions.
- **Enhancement:** AI refines its decisions by using new data and self-learning.

1.1 Entering the era of multiple AI model architecture

The evolution of artificial intelligence (AI) spans decades, transitioning from theoretical concepts to a transformative force across industries. In the 1950s, AI emerged with foundational ideas like Alan Turing's work on machine intelligence and the development of early rule-based systems. The 1980s and 1990s saw advancements in machine learning, driven by increased computational power and algorithms like neural networks, though limited by data and hardware constraints. The 21st century marked a turning point with the rise of big data, cloud computing, and deep learning, enabling breakthroughs in natural language processing, computer vision, and generative models. See Figure 1-1 for a timeline of this evolution.

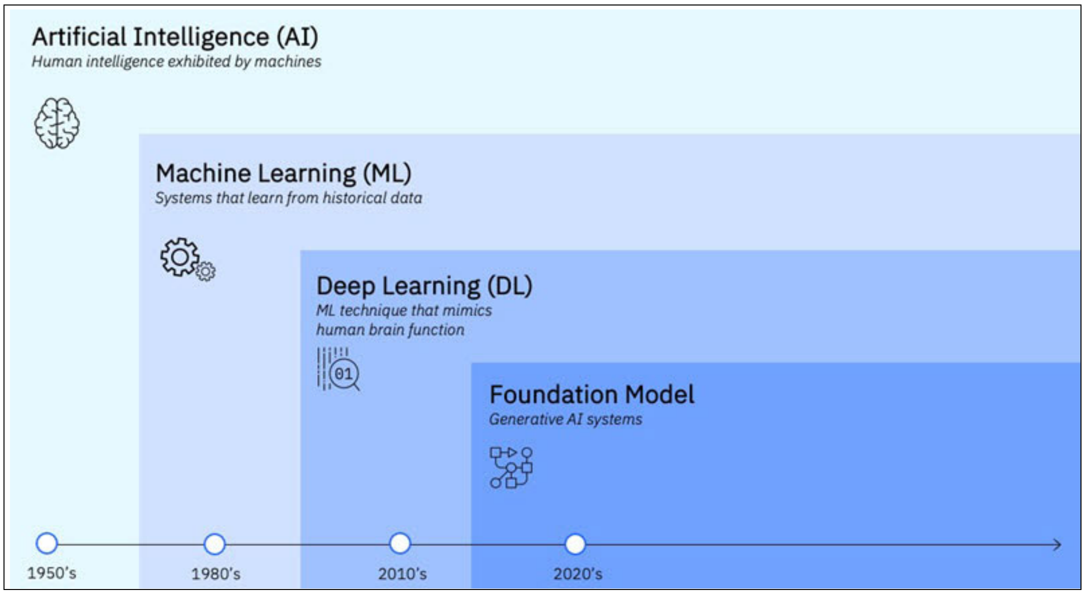


Figure 1-1 An evolution of AI technologies over time

Today, AI has progressed to include multiple AI model architectures. multiple AI model architectures can include both generative and predictive AI, as it is designed to handle diverse tasks by processing multiple data types such as text, images and audio. More discussion on this as well as a sample use case of a multiple AI model architecture can be found in "AI multiple model architecture" on page 33.

The AI landscape is expansive and encompasses several key families of algorithms, each contributing to its diverse capabilities, some of which are the following:

- Machine Learning (ML)

  ML-based systems are trained on historical data to uncover patterns. Users provide inputs to the ML system, which then applies these inputs to the discovered patterns and generates corresponding outputs. ML practical applications include predictive and classification tasks. For example, forecasting house prices in each area, or classifying business transactions for fraud.

  ML methods include supervised learning, semi-supervised learning, unsupervised learning and reinforcement learning.

- Deep Learning (DL)

  DL is a subset of ML, using multiple layers of neural networks, which are interconnected nodes, which work together to process information. DL is well suited to complex applications, like image and speech recognition.

- Generative AI (GenAI)

  AI model built using a specific kind of neural network architecture, called a transformer, which is designed to generate sequences of related data elements (for example, like a sentence). Generative AI can produce new content - text, audio, images or videos.

The introduction of Foundation Models, also known as Large Language Models (LLMs), marked a significant breakthrough for Generative AI. Pre-trained on extensive cross-industry datasets, LLMs can contain millions or even billions of parameters, which is reflected in their name. While their practical applications vary, including translation and image generation, this publication will focus on business applications of Generative AI.

LLMs introduce new challenges for AI infrastructure, requiring substantial compute resources for tasks such as deploying the model, training, fine-tuning, and model governance. Governance is essential because, over time, LLMs can hallucinate, producing grammatically correct content that lacks meaningful substance.

Combining both predictive and Generative AI technologies in business applications can yield deeper insights, improved accuracy and precision, and overall provide more robust and effective systems. Deploying multiple AI model solutions alongside business transactional systems can result in lower response times, which can be crucial for Generative AI use cases that can be slower due to the inherent architectural complexity of LLMs.

IBM z17 and LinuxONE 5 platforms have been designed to provide the necessary infrastructure for implementing a multiple AI model use cases.These platforms offer a sustainable environment designed for AI and generative AI business-critical workloads. Hardware acceleration, via the IBM Telum II co-processor or IBM Spyre™ AI accelerator, makes converged AI scalable and robust, dynamically scaling computing power to meet business SLAs.
Table 1-1 provides an overview that compares some of the features of Telum and Telum II.

Table 1-1 Telum and Telum II comparison chart

Feature	Telum	Telum II
On-chip AI acceleration	IBM z16® A01	IBM z17™ ME1[a]
Support for LLM compute primitives	No	Yes
Technology Node	7nm	5nm
Processor Cores	8 cores	8 cores
Capacity	200 characterizable processor units	208 characterizable processor units
Clock Speed	Over 5GHz	5.5GHZ
Cache	32MB L2 per core, 256MB virtual L3, 2GB virtual L4	36MB L2 per core, 360MB virtual L3, 2.88GB virtual L4
AI Acceleration	On-chip AI accelerator for inferencing	Designed for enhanced AI accelerator with 4x compute power
Data Processing Unit (DPU)	Not available	Integrated DPU for IO acceleration
Security	Transparent encryption of main memory	Designed for enhanced security features including quantum-safe methods

Feature	Telum	Telum II
Performance	Optimized for real-time analytics and decision-making	Designed for improved performance for AI-driven workloads and enterprise transactions

a. Calculations presented in this column were established through internal measurements. Your results may vary based on individual workload, configuration, and software levels. Visit the LSPR web page or *IBM z17 Technical Introduction*, SG24-8580 for more details.

Additionally, a rich software ecosystem is available, including various frameworks for end-to-end AI model development, training, inferencing, and governance. Model training can be done off-platform, with the model brought back in open standard formats (ONNX - Open Neural Network Exchange, PMML - Predictive Model Markup Language) for high-performance inferencing on the IBM Z platform.

1.2 Why use AI on IBM Z and LinuxONE?

Artificial intelligence (AI) demands significant computational power and relies on specialized architecture, making a strong and scalable infrastructure critical to unlocking its full potential. Without this foundation, developing and deploying state-of-the-art AI solutions is simply not viable. Infrastructure is integral throughout the AI lifecycle, with platforms like IBM Z and LinuxONE playing a key role in supporting mission-critical workloads and sensitive data. However, operationalizing AI, especially in environments with high-performance demands, presents unique challenges like the need for real-time insights while maintaining strict service level agreements (SLAs) and safeguarding data privacy.

A key emerging trend is the growing adoption of collocating AI models with critical business applications on IBM Z and LinuxONE. This strategy enables organizations to make faster and more informed decisions by processing data directly on-platform, which eliminates the need to move data elsewhere. By keeping AI processing close to where the data resides, businesses can enhance security, reduce latency, and accelerate the delivery of impactful outcomes.

Linux on IBM Z and LinuxONE platforms have been designed to offer several advantages for businesses looking to leverage AI in real-time transactions with large amounts of data:

- High Performance: These platforms are equipped with integrated accelerators for AI, such as the IBM Telum® processor, which significantly boosts the performance of AI inferencing. This allows businesses to process large volumes of transactions with low latency.
- Scalability: IBM Z and LinuxONE are designed to handle massive amounts of data, making them ideal for businesses that need to scale their AI applications. They can manage data across platforms efficiently, ensuring seamless integration and high throughput.
- Security and Compliance: The AI Toolkit for IBM Z and LinuxONE includes IBM Elite Support and IBM Secure Engineering, which help ensure that AI frameworks are secure and compliant with industry regulations.
- Cost-Effective: These platforms offer cost-effective entry points for using AI, reducing costs and complexity while accelerating time to market with lightweight tools and runtime packages.

- Power Efficiency: These platforms have been designed to enhance power efficiency by consolidating Linux workloads, which can reduce energy consumption . Additionally, these systems can help lower CO2 emissions, contributing to more sustainable IT operations.

Chapter 2., “Optimized Model Serving Solutions” on page 9 will highlight two key products that will empower AI innovation. AI Toolkit for IBM Z and LinuxONE is designed to make it easier to use popular open-source AI frameworks such as TensorFlow and PyTorch directly on the platform. IBM Cloud Pak® for Data, which runs on both Linux on IBM Z and LinuxONE, delivers a cloud-native platform for managing AI and analytics, enabling organizations to efficiently build, deploy, and govern AI models directly where their data resides. The overall AI goal is to enable businesses to make quicker, smarter decisions without the complexity of data movement.

AI is making databases smarter and more efficient. IBM Db2 AI for z/OS uses machine learning to enhance query performance by optimizing how data is accessed and processed. With AI driven tools such as the IBM watsonx® Assistant™ and IBM watsonx Code Assistant™ tasks can now be automated, interactions may be simplified, and application modernization can now be accelerated on IBM Z and LinuxONE

IBM Z and LinuxONE offer a dedicated software ecosystem and full-stack hardware architecture, forming a comprehensive platform for AI acceleration, described in section 1.4, “Integrated Accelerator for AI” on page 6. The Telum and Telum II processors feature an on-chip AI accelerator, enabling high-speed AI inferencing. For even more demanding workloads, the Spyre™ accelerator provides additional AI processing power. IBM Z and LinuxONE can handle billions of AI inference requests daily, delivering real-time AI on an enterprise scale.

1.3 IBM Z and LinuxONE ecosystem overview

IBM Z and LinuxONE offer operating system and virtualization options that include IBM z/OS, IBM zCX, Ubuntu, SUSE, and Red Hat OpenShift. These selections offer flexibility for running containerized workloads, integrating with cloud-native environments, and supporting modern Dev Ops practices.

In this section, we provide an overview of the ecosystem which also includes a wide range of programming languages, libraries, and compilers, such as Python, Java, Go, GCC, and C++. Optimized libraries like OpenBLAS and Deep Learning Compiler further accelerate AI computations, ensuring efficient execution of workloads.

For IT operations, the platform offer IBM Db2 AI for z/OS, IBM Z Anomaly Analytics, and IBM Cloud® Pak for AIOps, along with Instana® for real-time monitoring and observability. These tools help optimize system performance, detect anomalies, and improve operational efficiency.

Key AI and analytics capabilities are available on IBM Z and LinuxONE. These include IBM Machine Learning for z/OS, the Python AI Toolkit, IBM Synthetic Data Sets, Cloud Pak for Data, AI Toolkit for IBM Z and LinuxONE and IBM Db2 with SQL Data Insights, which enable organizations to leverage AI and machine learning for data driven decision making. Db2 Analytics Accelerator and IBM Watson® further enhance the ability to analyze huge amounts of structured and unstructured data efficiently.

IBM Z and LinuxONE support a variety of AI and data processing frameworks, including PyTorch, TensorFlow, Apache Spark, Scikit-learn, and Anaconda. These frameworks enable advanced machine learning, deep learning, and big data analytics, making it easier to deploy AI workloads on the platform.

IBM Z and LinuxONE are widely used across multiple industries, including banking, finance, insurance, retail, hospitality, healthcare, and government. These sectors benefit from the platform's enterprise-grade security, high availability, and scalability.

1.4 Integrated Accelerator for AI

At the heart of IBM z17 is the Telum II processor and the new on-chip Integrated Accelerator for AI. The integrated AI accelerator uses dedicated silicon on the chip and is designed to provide extremely high performance and consistent low latency inferencing for processing a mix of traditional transactional and AI workloads at speed and scale.

With the on-chip AI accelerator, complex AI inferencing leveraging real-time data can be integrated into transactional workloads and deliver the insights needed for every transaction while still meeting even the most stringent SLAs.

The IBM z17 and IBM LinuxONE Emperor 5 have been designed to process up to 5 million inference operations per second or 450 billion inference operations per day with less than 1 ms response time using a Credit Card Fraud Detection Deep Learning model.

1.5 Migrate AI application to IBM Z and LinuxONE

Some reason to migrate your AI applications to Linux are as follows:

- IBM Z and LinuxONE offer a highly available, resilient environment with built-in fault tolerance, ensuring AI applications run with minimal downtime.
- The consolidation of workloads are enabled, which can lead to better cost efficiency by reducing the need for multiple physical or virtual servers.
- AI into core business processes becomes seamless, with lower latency and enhanced security.

IBM Z and LinuxONE have the flexibility to deploy AI across hybrid cloud environments. Whether on-premise or in the cloud with providers such as AWS, Azure, and IBM Cloud, businesses can train and run AI workloads anywhere, making AI deployment frictionless and scalable.

Figure 1-2 provides a migration example. On the left side of the figure, an AI application interacts with TensorFlow, running on Linux, which is hosted on a distributed system. This setup relies on multiple servers or a cloud-based infrastructure, which can lead to encounters in security, scalability, and operational management. AI workloads often require substantial computing power, which results in high operational costs and added complexity in managing multiple nodes.

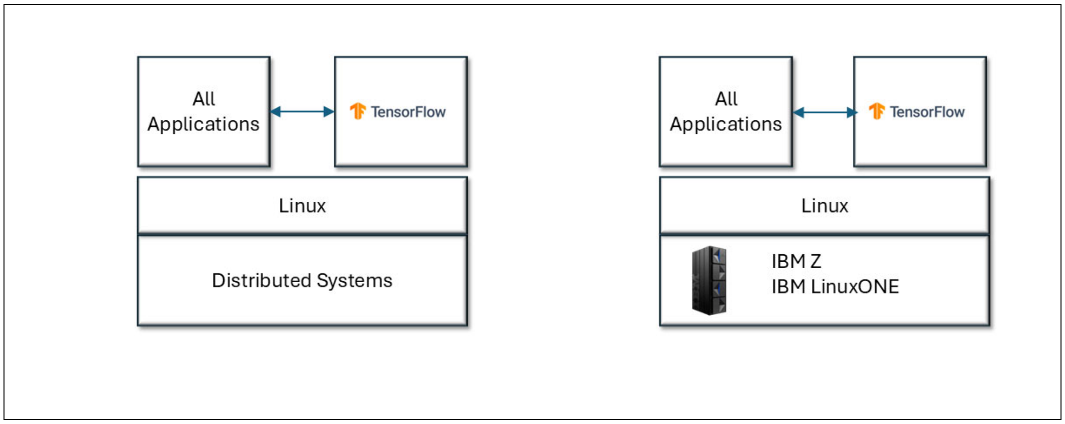


Figure 1-2 Migrate AI application to IBM Z and LinuxONE

On the right side, we see the AI application on IBM Z and LinuxONE. The AI application and TensorFlow continue to run on Linux, but instead of being hosted on a distributed system, they now operate on Linux on IBM Z or LinuxONE. This shift provides key advantages, including enhanced security features like pervasive encryption and Secure Execution, ensuring AI workloads remain protected. The IBM Z AI accelerator capabilities help improve efficiency and performance, making AI inference and model training faster and more reliable.

1.5.1 Migrate AI application to Red Hat OpenShift, IBM Z and LinuxONE

As shown in Figure 1-3, AI applications migrated to Red Hat OpenShift on IBM Z and LinuxONE, can deliver a more scalable, secure, and efficient deployment model. Scaling Linux environments on distributed systems often require provisioning multiple servers, leading to complexities in workload management, latency, and overall reliability.

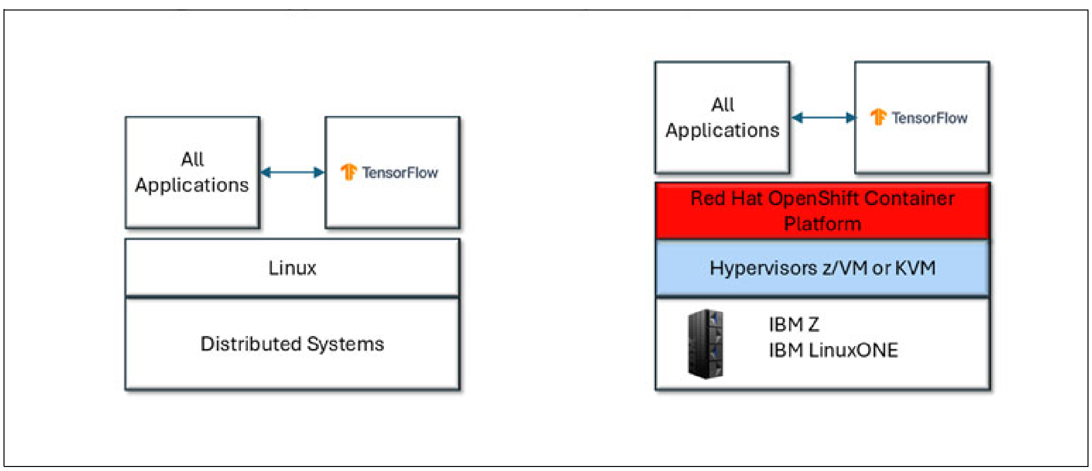


Figure 1-3 Migrate AI application to IBM Z and LinuxONE with Red Hat OpenShift

In an architecture application that is migrated to IBM Z and LinuxONE with Red Hat OpenShift, instead of running directly on Linux, the AI application and TensorFlow now operate within containers managed by the Red Hat OpenShift Container Platform. This setup is further optimized with hypervisors such as z/VM or KVM which provide proficient resource allocation and virtualization. By leveraging OpenShift AI, workloads can benefit from container orchestration, automated scaling, and improved deployment across environments.

By moving AI workloads to IBM Z and LinuxONE with Red Hat OpenShift:

- IBM Z and LinuxONE can offer a highly available, resilient environment with built-in fault tolerance, ensuring AI applications run with minimal downtime.
- Infrastructure management is simplified and security is enhanced, as IBM Z and LinuxONE are designed to provide industry-leading encryption and isolation capabilities.
- Running AI workloads in containers ensures better portability and agility, making it easier to integrate with modern Dev Ops workflows.
- With OpenShift, organizations can dynamically scale AI applications based on demand while reducing the overhead of managing individual virtual machines or physical servers.

IBM Z and LinuxONE enable seamless AI integration into enterprise workloads across industries such as banking, finance, healthcare, retail, and government. AI models can be built and trained anywhere by using popular frameworks such as Python, Jupyter, TensorFlow, PyTorch, Keras, and Snap ML. These models are then optimized using ONNX and deployed efficiently on IBM Z hardware through IBM Snap ML or the Deep Learning Compiler and operating systems such as z/OS, Linux, and Red Hat OpenShift.

2

Optimized Model Serving Solutions

In this chapter, we describe the following solutions offered for optimized model serving

2.1 AI Toolkit for IBM Z and LinuxONE

AI Toolkit for IBM Z and LinuxONE is a collection of popular Machine Learning and Deep Learning serving solutions available from the IBM Z and LinuxONE Container registry with IBM Elite support and can be used as a bundle or individually, depending on your needs.

AI Toolkit for IBM Z and LinuxONE has six high performing ML and DL frameworks in the form of TensorFlow, TensorFlow Serving, SnapML, Triton Inference server, PyTorch and Deep Learning compiler.

Note: For the most recent information about AI Toolkit for IBM Z and LinuxONE, see: https://www.ibm.com/products/ai-toolkit-for-z-and-linuxone

These frameworks and serving solutions are optimized to leverage the Telum on-chip AI accelerator which has been designed to be highly efficient for faster inference and scoring of traditional AI for both ML and DL.

AI Toolkit provides enterprise grade IBM Elite Support which can help bring solutions based on open source into production on an enterprise system.

Apart from service and support, IBM has put these open-source AI libraries through a secure engineering process, scanned and vetted them, and constantly monitors them for vulnerabilities.

For more information, see section 3.1.4, “AI acceleration” on page 26.

Figure 2-1 provides an overview of the high level architecture of AI Toolkit for IBM Z and LinuxONE.

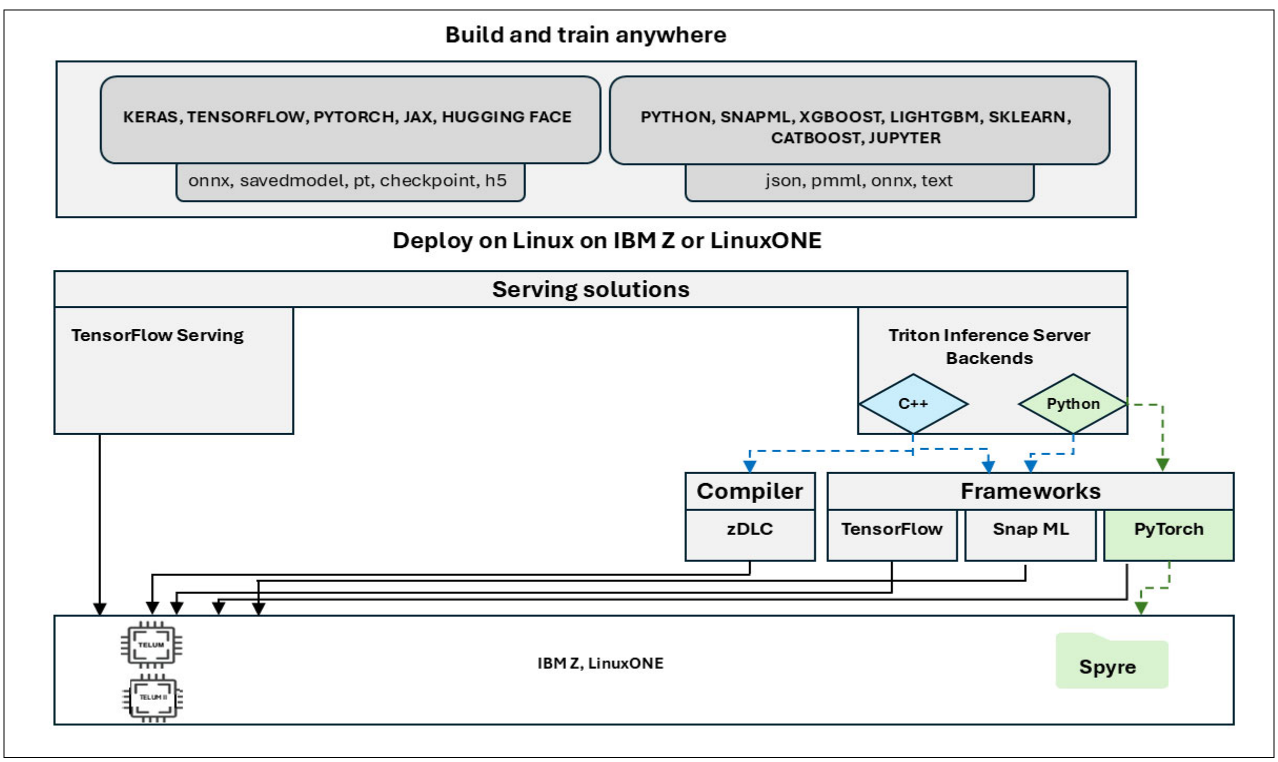


Figure 2-1 AI Toolkit for IBM Z and LinuxONE High Level Architecture

The AI Toolkit architecture is designed to provide flexibility during model development and efficient deployment on IBM hardware such as the IBM Z and LinuxONE with the on-chip accelerators.
In Figure 2-1, we show the following layers:

1. The Build and Train Anywhere layer.
    - Supports mainstream libraries and frameworks including: Keras, TensorFlow, PyTorch, JAX, and Hugging Face for deep learning.
    - Python, Snap ML, XGBoost, LightGBM, scikit-learn, CatBoost, and Jupyter for traditional ML and experimentation.

   – Model artifacts export in standard formats: ONNX, SavedModel, pt (PyTorch), checkpoint, h5, and structured data formats like JSON, PMML, and text.

   – Models get deployed with serving solutions provided on the IBM Z platform:

    - TensorFlow Serving directly targets the TensorFlow framework.
    - Triton Inference Server (IS) supports C++ and Python backends, which extend compatibility.

2. Deploy: The following serving options target optimized AI frameworks:

   – TensorFlow

   – Snap ML: IBM's high-performance machine learning library

   – zDLC: The IBM Z Deep Learning Compiler

   – PyTorch: Integrated via a Python backend and using Triton IS

   The serving stack is deployed on IBM z16 or z17 systems based on the Telum chip, which provides on-chip AI inferencing.
   The frameworks are tuned to take advantage of this hardware for low latency and high throughput AI inference.

On the bottom right of Figure 2-1, there is SPYRE, which is an application integration layer or interface that allows business applications and business logic to ”talk” to AI models natively, bridging the gap between data, production systems, and models.

This architecture supports complete AI lifecycle management from training with any toolset of choice to optimized deployment on Linux. It leverages open source flexibility, high-performing serving and enterprise-level performance that is driven by IBM hardware and AI acceleration.

The frameworks make up the AI Toolkit for IBM Z and LinuxONE and deployment options are shown in Figure 2-2 and are discussed further in the coming subsections.

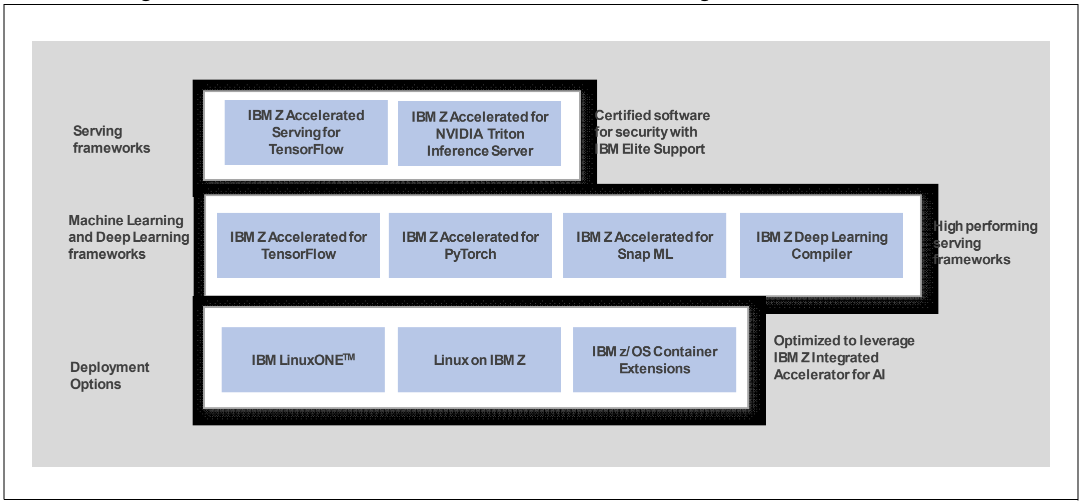


Figure 2-2 Frameworks and deployment options in AI toolkit for IBM Z and LinuxONE

Note: All of the serving, machine leaning and deep learning frameworks are accelerated for both IBM Z and LinuxONE.

2.1.1 IBM Z Accelerated for TensorFlow

Using TensorFlow provides open source deep learning tools and a framework for data scientists. Data scientists can now leverage Machine Learning (ML) and Deep Learning (DL) libraries. TensorFlow can be used for several use cases including Natural Language Processing (NLP)/text-based applications, image recognition, voice search and many more. The system operates in the Facebook DeepFace image recognition platform along with Apple's Siri voice recognition service. TensorFlow provides a flexible, scalable system for defining, training, and deploying machine learning models by managing data flow and computations with robust support for deep learning and hardware optimization.

TensorFlow enables infusion of DL & ML into mission critical workloads at scale leveraging Integrated Accelerator for AI.

TensorFlow can be accessed in the following ways:

1. Linux distribution on zCX (z/OS Container Extensions) and Linux on IBM Z and LinuxONE. Also, distribution channels like Docker Hub etc.
2. IBM Cloud Pak for Data v4.6 and later.
3. IBM LinuxONE container image repository

Options 2 and 3 are designed to provide optimal performance and security validation.

The model powered by TensorFlow, shown in Figure 2-3:

1.Receives input data from different classes (clusters) of data (classes of data are shown in Figure 2-3 as ω1, ω2 and ω3).

2.Learns to distinguish between classes using training.

3.Resulting outputs are predictions or inferences based on input data such as:

– Classification: Probabilities or class labels (for example, [0.9, 0.1] for a binary classifier or [“cat”] for image recognition).
– Regression: Numerical values (for example,[42.5] for a house price prediction).
– Generative Models: Data like images, text, or audio (for example, a generated image tensor from a Generative Adversarial Network or GAN).

.

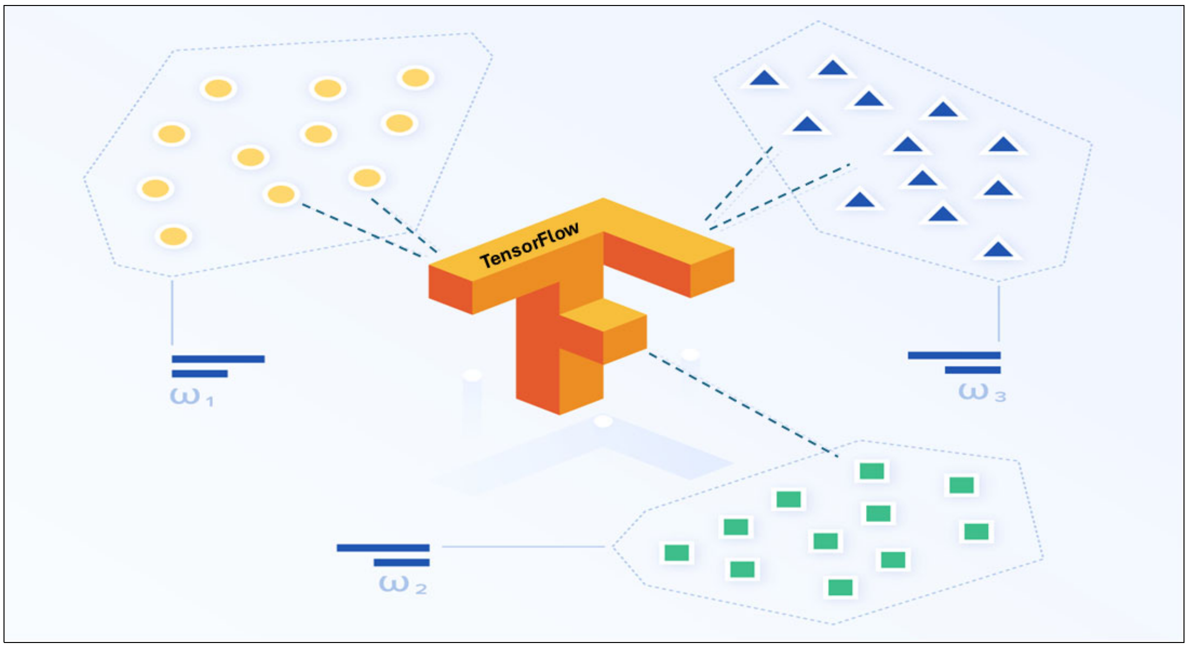


Figure 2-3 TensorFlow with Linux

2.1.2 IBM Z Accelerated for TensorFlow Serving

In this section, we discuss TensorFlow Serving and how to deploy a model created with TensorFlow for inferencing. TensorFlow Serving is a library for serving machine learning models developed with TensorFlow and provides a rich set of features among which are model versioning, automatic batching of requests, server side and client side batching.

Predictions from machine learning models are served via HTTP REST or Google Remote Procedure Call (gRPC) interfaces. This supports high throughput, low latency inference serving.
Figure 2-4 presents a machine learning model deployment that uses TensorFlow Serving.

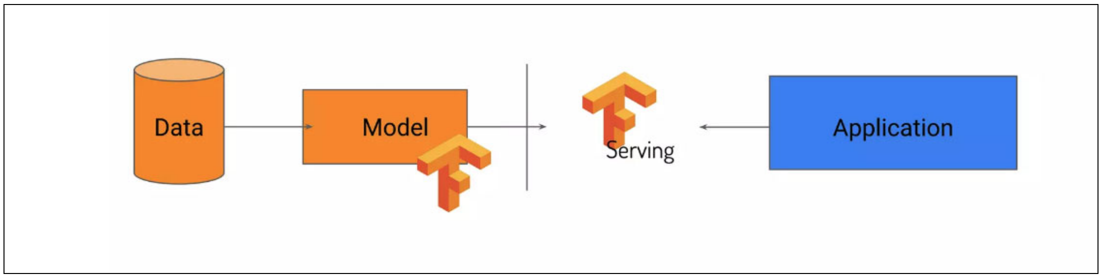


Figure 2-4 Machine learning model deployment using TensorFlow Serving

The following is a high-level overview of ML model deployment that uses TensorFlow Serving:

1. Train your model on data
2. Export it using TensorFlow
3. Deploy it with TensorFlow Serving
4. Integrate it into a real-world application for inference.

Exclusive to TensorFlow are static computation graphs that serve production scalability and deployment needs along with operational efficiency. The flexible debugging capabilities of PyTorch (discussed further in "IBM Z Accelerated for PyTorch" on page 16) match its strengths in ease of use yet TensorFlow is more popular in business settings because it has established its tools TensorFlow Serving and TensorFlow Lite for enterprise and mobile applications.

TensorFlow is integrated into various IBM platforms such as Watson Machine Learning and Cloud Pak for Data. These solutions enable companies to create and implement AI models, and TensorFlow is commonly employed to predict factory equipment failures (predictive maintenance), detect fraud within the banking sector, analyze various medical images, power chatbots, and perform vision-based tasks in retail and security. Most businesses use TensorFlow for predictive modeling and other deep learning tasks. It serves real-world purposes, offering tailored solutions to sophisticated problems faced by large-scale enterprises.

2.1.3 IBM Z Accelerated for Snap ML

Snap ML is an open-source library being developed and maintained by IBM Research® that supports acceleration of training and inference of popular machine learning models

It provides high-performance implementation for:

- Generalized linear models
- Tree-based models
- Gradient boosting models

SnapML supports the IBM Z Integrated AI Accelerator in Telum and II and is packaged along with AI Toolkit, IBM Cloud Pak for Data (which runs on both Linux on IBM Z and LinuxONE), Machine Learning for IBM z/OS and available as standalone container images via PyPi.

For more information on this, see: https://github.com/IBM/ibmz-accelerated-for-snapml

For some examples of notebooks that demonstrate how to use the IBM Snap Machine Learning (Snap ML) library, see: https://github.com/IBM/snapml-examples

Snap ML supports both linear and tree-based machine learning algorithms. Figure 2-5 shows some of the learning algorithms.

Figure 2-5 Linear and tree-based machine learning algorithms

2.1.4 IBM Z Deep Learning Compiler (zDLC)

zDLC enables you to build and train deep learning models using your choice of open-source frameworks such as TensorFlow or PyTorch. zDLC supports C++, Java and Python APIs.

You would compile deep learning models that are optimized to leverage Integrated Accelerator for AI in order to deploy and run on IBM LinuxONE 4 and IBM z16 and z17.

The compiled shared library files can be embedded in applications independent of the original framework's dependencies and packages.

For more information on zDLC, see: https://github.com/IBM/zDLC

2.1.5 IBM Z Accelerated for NVIDIA Triton Inference Server

The NVIDIA Triton Inference Server is an open-source software that helps with AI model deployment and execution. It is a high performing model inference framework that was optimized by the AI on IBM Z team for the IBM Z and LinuxONE architecture.

Triton has the ability to create custom model backends which makes it very flexible. As a part of AI Toolkit for IBM Z and LinuxONE the focus is on the following backends:

1. Traditional machine learning models in the PMML, ONNX, or JSON format that are run with an IBM Snap ML runtime.
2. Deep Learning models in the ONNX model format and compiled with the IBM Deep Learning Compiler.

Key Features of the Triton Inference Server include:

- A high-performance inference server that supports the deployment of ML or DL models at scale.
- Support for multiple frameworks.
- Supports model ensembles

- Concurrent model execution along with dynamic batching.
- Assists in meeting strict customer SLAs.
- The ability to download the Triton container from the IBM Cloud Container Registry.

The Triton Inference Server is designed to run predictions of customer transactions 3.5x faster by co-locating an application with the Snap ML library on IBM LinuxONE Emperor 4 than running predictions remotely using the NVIDIA Forest Inference Library on a compared x86 server. Figure 2-6 demonstrates the co-location of an application with the Snap ML library on IBM LinuxONE vs when not co-located.

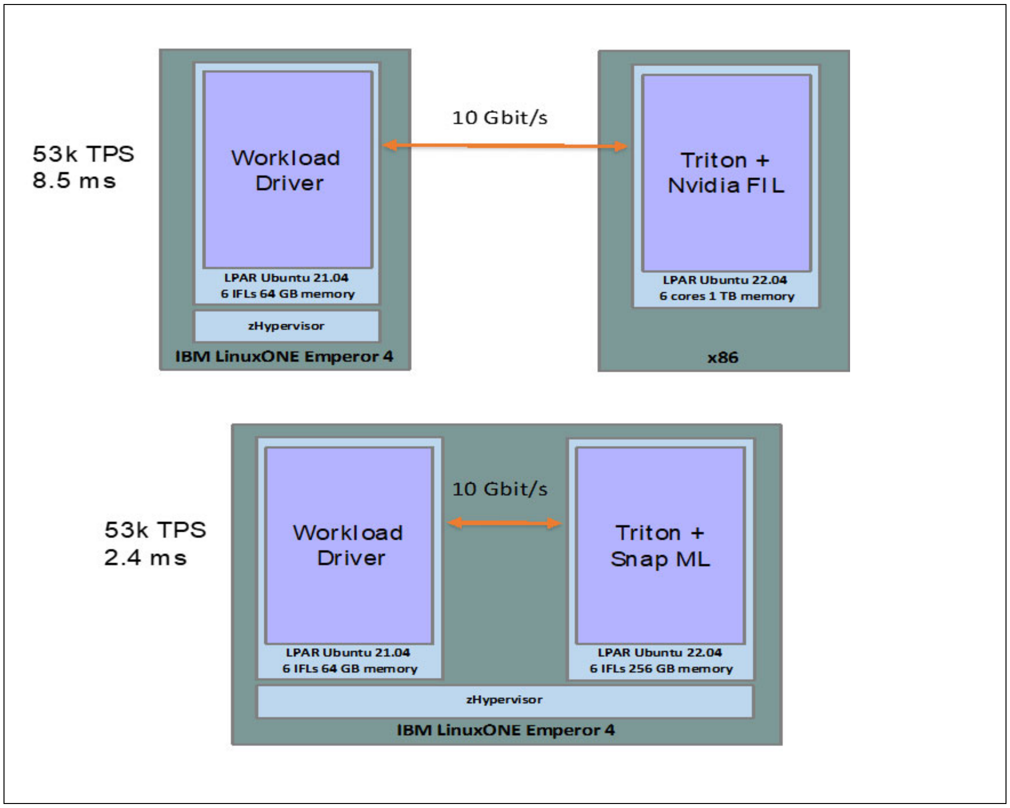


Figure 2-6 Comparison of AI Inference Deployment: IBM LinuxONE with Snap ML vs x86 with NVIDIA FIL

Data Pre-processing and Inference together using low level programming languages can reduces the overall time to inference is KEY to meet strict SLAs

2.1.6 IBM Z Accelerated for PyTorch

PyTorch supports Linux environments, including the s390x architecture used by IBM LinuxONE. Official PyTorch releases (e.g., via pip or conda) include builds for s390x, and IBM has actively contributed to ensuring compatibility, especially for enterprise workloads. PyTorch-based AI applications take advantage of other IBM Z and LinuxONE features, including the encryption of data in transit, at rest, and in use with confidential computing.

PyTorch is a fully featured framework for building deep learning models and is designed to provide flexibility and high speeds for deep neural network implementations.

PyTorch on Linux for IBM Z enables infusion of DL and ML into mission critical workloads at scale leveraging Integrated Accelerator for AI. It is available from the IBM LinuxONE container image repository.

Key Benefits of PyTorch with Linux include:

- Supports both traditional AI and Foundation model stacks on IBM Z.
- Enables deployment of Transformers and encoder-based models as well as Ensemble AI models.
- Optimized for IBM Telum, IBM Telum II and Spyre.
- It has the ability to import pre-trained models and leverage AI Accelerators for faster deployment

The dynamic computation graph of PyTorch provides users an intuitive framework. AI-based research and experimentation are best served by PyTorch, especially with language understanding using Hugging Face Transformers. Medical AI, federated learning, and generative models like AI-based image and text generation are evolving fields that require swift prototyping of new concepts. way.

Figure 2-7 shows the architecture that supports the seamless infusion of AI/ML capabilities within enterprise-class transactional systems and enhances reuse and leverage of existing IBM infrastructure. This architecture is optimized for performance, flexibility, and enterprise scalability, leveraging open-source frameworks and IBM-optimized toolkits.

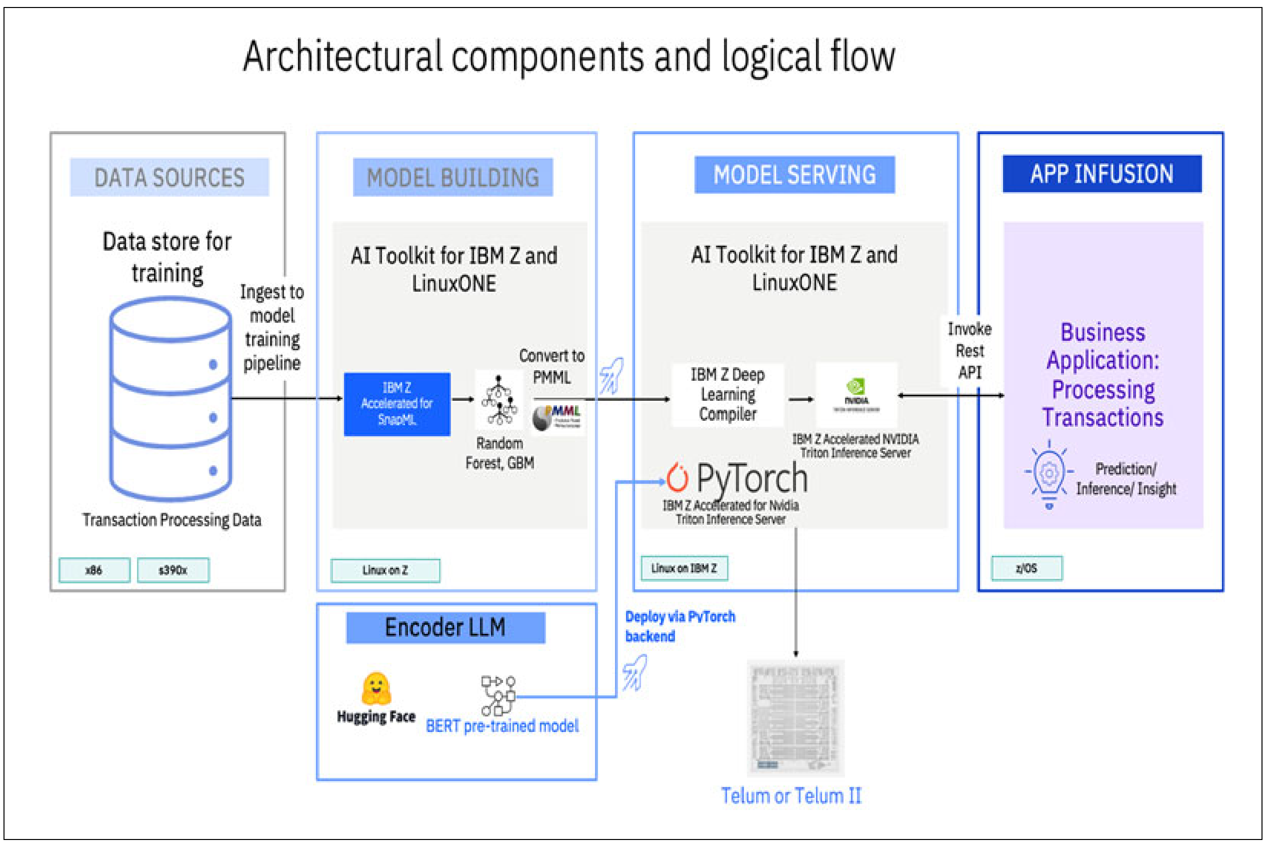


Figure 2-7 End-to-end architecture for AI integration in business applications with Linux

The data flow consists of data ingestion, training and deployment of models, serving, and application integration.

The following is a description of the sections shown in Figure 2-7.

1. Data Sources
   In our example, a training data store containing transaction processing
   data containing transaction processing data, The data sources may be derived from x86 or s390x platforms. The information is fed into the pipeline of model training for building AI models.
2. Model Building

   During the Model Building phase, we deploy the AI Toolkit for IBM Z and LinuxONE. Here, IBM Z optimized Snap ML
   Algorithms such as Random Forest and Gradient Boosting Machines (GBM) are trained and subsequently exported to PMML (Predictive Model Markup Language) for standardization and portability.
   In addition, Encoder LLMs (such as Hugging Face pre-trained BERT models) are also supported, particularly convenient for natural language processing.

   These models are served through the PyTorch backend on IBM Z infrastructure.

   AI quantization can be applied during or following model training.
   This method decreases the accuracy of numerical representations—for example, from 32-bit floats to 8-bit integers—without compromising the integrity of the data blocks.
   The following are the advantages:
   1. Increased computational speed
   2. Lowered memory footprint
   3. Decreased energy consumption
   These gains are essential when running models on resource-limited environments or high-volume enterprise systems such as IBM Z.
3. Model Serving

   In Model Serving, the trained model is compiled with the IBM Z Deep Learning Compiler. The models are served by the NVIDIA Triton Inference Server, which is accelerated on IBM Z.
   This allows for high-performance inference with PyTorch as the backend for deployment. The compute platform is Telum or Telum II, optimized for AI inference workloads on IBM Z.
4. App Infusion

   Finally, in App Infusion, the models are infused with business applications that handle transactions. This interaction occurs through REST APIs, which call the inference engine in real time.

   The applications are z/OS-based and use AI models to generate insights, predictions, and inferences to support business decision-making and operational efficiency.

For some use case examples that leverage AI Toolkit for IBM Z and LinuxONE, see sections 4.2, “Use case: An advanced multiple AI model framework for anti-money laundering” on page 40 and 4.3, “Use case: Credit risk assessment - Triton Inference Server (TIS) and Open-Source Framework ” on page 43

In this section, we provide an overview of two key use cases that provide an example of the topics discussed in this chapter.

2.2 IBM Cloud Pak for Data

IBM Cloud Pak for Data is a cloud-native Data and AI platform that helps modernize data management, analytics, and AI to help drive outcomes for business faster. It takes advantage of IBM Z and LinuxONE features such as pervasive encryption, high availability, and

virtualization (via KVM or z/VM) and are designed to ensure robust data management, analytics, and AI workloads. This combination enables organizations to modernize data strategies with enhanced security and efficiency, particularly for mission-critical applications in industries like finance and healthcare.
IBM Cloud Pak for Data offers significant advantages, such as leveraging the advanced Telum and Telum II embedded accelerator chip and the mainframe quality of service to meet enterprise customers demands such as high performance, scale, reliability, and unmatched security.

> **Note:** For more information about IBM Cloud Pak for Data see:
> https://www.redbooks.ibm.com/redpapers/pdfs/redp5695.pdf
> https://www.ibm.com/products/cloud-pak-for-data

2.2.1 Overview of services available

With Cloud Pak for Data running on Linux, users have the ability to:

1. Collect, store, ingest, and access data.
2. Prepare and govern data.
3. Run analytics, data science, and machine learning techniques on data.
4. Build, train, deploy, and service models.
5. Automate and monitor the model lifecycle.
6. Use and run open-source capabilities such as TensorFlow and SnapML.

For more information on the services available with the latest IBM Cloud Pak for Data, see https://www.ibm.com/docs/en/software-hub/5.1.x?topic=requirements-planning-software-hub-z-linuxone

Figure 2-8 shows all the available components and services on IBM Cloud Pak for Data.

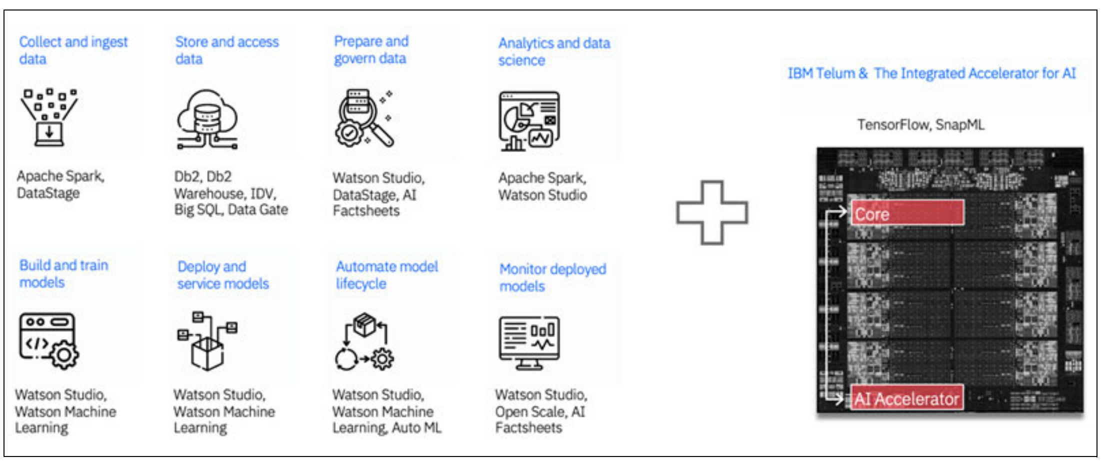


Figure 2-8 Available components and services on IBM Cloud Pak for Data

2.2.2 Getting Started with Db2 Data Gate

IBM Data Gate with IBM Cloud Pak for Data provides a modern hybrid cloud approach for cloud access to data originating on IBM Z. It provides an end-to-end solution to ensure data is synchronized for hybrid cloud, analytics, and AI initiatives with reduced cost and effort.

Db2 Data Gate provides an end-to-end solution to ensure that Db2 for z/OS data is available and synchronized from sources on IBM Z to targets optimized on IBM Cloud Pak for Data. A log data provider captures log changes from Db2 for z/OS and sends consolidated, encrypted changes to a log data processor residing on IBM Cloud Pak for Data. These changes can be sent to one or more Db2 Data Gate targets, reducing the complexity and cost of application development. The log capture processing is fully IBM z Integrated Information Processor (zIIP) enabled to ensure that there is little impact to general processing on IBM Z. Targets, such as Db2 Advanced or Db2 Warehouse, are determined based on consuming application requirements and are fully optimized to ensure low latency and high throughput.

Figure 2-9 shows the end-to-end flow of how Db2 z/OS data can be integrated and synchronized to IBM Cloud Pak for Data on the IBM Z architecture.

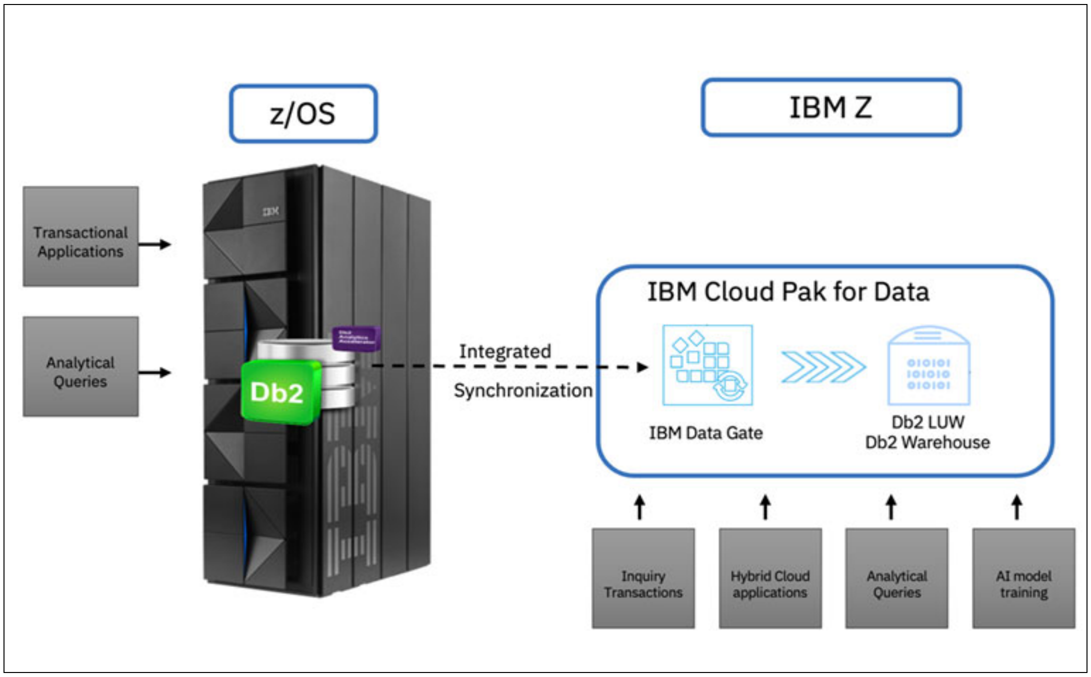


Figure 2-9 End-to-end flow

For more information on this topic, see Chapter 4, "Bringing it all together with the use cases" on page 37.

2.2.3 AI governance

AI governance refers to the frameworks, policies, and processes that ensure the ethical, transparent, and responsible development, deployment, and use of artificial intelligence systems. It encompasses guidelines for managing AI risks, ensuring fairness, accountability, and compliance with regulations, while promoting trust and aligning AI outcomes with organizational and societal values.

In this section we provide an overview of three AI governance tools that can play a crucial role in protecting data.

AI Factsheets

AI Factsheets are a fundamental building block of an AI governance solution, available in Cloud Pak for Data and IBM watsonx.governance®. They are designed to capture data throughout a model's lifecycle, providing stakeholders with the information they need to evaluate that model. Additionally, AI Factsheets increase transparency for auditors and regulators. AI Factsheets catalog the data that is used to train the model, the frameworks and runtimes used in its development, and the metrics gathered during its evaluations. The models are categorized by use case, and organized by lifecycle stage (Develop, Test, Validate, or Operate). If a model is performing below its configured thresholds, an alert is displayed as well, allowing the business to see at a glance the status of all models associated with a use case. AI Factsheets are also easily exportable to a wide variety of formats, such as PDF files, to support a wide variety of stakeholders.

With AI Factsheets on IBM Z, organizations can ensure model governance and compliance by tracking and documenting the detailed history of AI models along with ensuring compliance and providing visibility to all stakeholders. Organizations also need to monitor models in production to ensure that the models are valid and accurate and are not introducing bias or drift from the intended goal. This can be achieved by managing risk and ensuring responsible AI through the monitoring of deployed AI models using Watson OpenScale.

Watson Openscale

IBM Watson Openscale is another fundamental building block of an AI governance solution, which is also available in Cloud Pak for Data and watsonx.governance. Watson Openscale supports the monitoring, evaluation, and analysis of AI with trust and transparency to help understand how AI models make decisions. It leverages an enterprise-grade environment for AI applications that provides visibility into how your AI is built and used. It enables businesses to operate and automate AI at scale with transparent and explainable outcomes. Additionally, Watson Openscale allows organizations to monitor models in production to ensure that models are valid and accurate and not introducing bias or drifting away from intended goals. This increases the quality and accuracy of AI model predictions allowing you create high quality models. Together with AI Factsheets and Watson Openscale, organizations can manage risk and ensure responsible AI.

watsonx.governance

watsonx.governance is a governance toolkit designed to direct, manage, and monitor artificial intelligence (AI) activities within an organization. watsonx.governance has governance focus in three key areas:

1. Lifecycle Governance - Monitor, catalog and govern AI models from anywhere, throughout the AI lifecycle.
2. AI Risk and Security Management - Proactively identify and manage risks by automating workflows to ensure accountability and ownership of controls associated with those risks.
3. Regulatory Compliance - Assist clients adhere to various regulations by translating regulations into sub-mandates and policies.

With watsonx.governance, organizations and clients can ensure AI governance to manage risk and compliance cross the entire AI lifecycle.

3

Building enterprise ready AI applications at scale

Enterprises have business-oriented requirements for their applications for supporting their Service Level Agreement (SLAs). Artificial intelligence (AI) applications are no exception. Reliability, security and scalability are key characteristics for enterprise-ready AI applications and their underlying infrastructure to operate effectively at scale.

AI application at scale means that the deployment and operation of AI systems across large, complex, or widespread environments to deliver significant impact, efficiency, or value. It involves moving beyond small-scale experiments or proofs of concept (PoCs) to implementing AI solutions that handle high volumes of data, users, or processes in real-world settings, often across an organization or industry.

This chapter describes the values of Linux in the context of AI applications. We guide you through the full end-to-end process of building production-ready AI solutions at scale and conclude the chapter with a sample of key-market use cases, supported on Linux.

3.1 Reasons to deploy AI solutions on Linux on IBM Z and LinuxONE

As described in Chapter 1, “Introduction: The convergence of AI and enterprise Linux systems” on page 1, Linux enables enterprises to run Linux-based mission critical workloads in a reliable, secure and scalable environment.

LinuxONE offers an infrastructure, dedicated to enterprise Linux workload, by leveraging IBM Z infrastructure core values.[1]

Those core values - reliability, availability, serviceability, performance and scale - are fully exploited by modern AI applications. AI applications, deployed on LinuxONE platform, adhere to enterprises' strict SLAs and business regulations.

Another key value is an attractive TCO (Total Cost of Ownership) for running AI workloads makes the IBM Z and LinuxONE an economical choice as well. Linux and specifically AI workloads need hardware resources to sustain the demand, and it's important to incorporate the Total Cost of Ownership (TCO) when making the infrastructure decision.

For AI-intensive workloads, the IBM Z and LinuxONE hardware platform provides an innovative on-chip AI processor, dedicated for streamlining CPU-hungry AI operations. The co-processor is present on every IBM Z and LinuxONE computing chip, and transparently to AI applications, it accelerates deep learning and predictive AI algorithms and operations. This is designed to be energy efficient and effective for any business use case in any industry - scoring a business transaction for fraud or AML (Anti-Money Laundering) or processing a medical image recognition.

The latest generation of IBM Z and LinuxONE builds on the strong foundation of Telum technology and expands AI with an introduction of Telum II and the optional Spyre cards, supporting multiple AI model composite solutions - a mix of AI and GenAI models.

Data is at heart of any AI solution, being the source for model development and training. LinuxONE inherits IBM Z high-performance, data-driven architecture for efficient and more streamlined data serving. LinuxONE can deploy data serving solutions (data lakes, databases - SQL and non-SQL) and bring AI solutions closer to the data to minimize the latency and thus improve the model's inference time.

AI solutions on the LinuxONE platform can be embedded into business transactional applications, systems of the record, integrated with data serving and middleware products.

See Chapter 2, “Optimized Model Serving Solutions” on page 9 for a description of data-serving solutions available on the platform.

While IBM Z and LinuxONE provide a secure, high-performing and scalable infrastructure for AI solutions, it also supports the latest and greatest AI software ecosystem, enabling AI and data scientists with rich tooling for AI application development, testing, deployment and governance. For high-performance AI model inferencing, clients have a choice between IBM enterprise software or open-source frameworks that were optimized for the IBM Z and LinuxONE architectures.

When choosing the right infrastructure for AI production-ready solutions deployment, consider IBM Z or LinuxONE, which that adheres to the following requirements:

- Availability - serving data and AI solutions 24/7

1 https://www.ibm.com/linuxone

- Reliability - serving data and AI solutions with 99.9999999%
- Scalability - scaling out (horizontally and vertically) AI solutions dynamically following the demand curve
- Performance - minimizing the latency and inferencing time to adhere to the SLAs, fully leveraging built-in AIU co-processor and software ecosystem
- Security - AI-based solutions operate on the data that can be commercially sensitive and securing the data, applications and models is key for a production-ready environment.
- TCO - providing all the above in an economical and sustainable manner.

3.1.1 Reliable and available

IBM Redbooks publication *Leveraging LinuxONE to Maximize Your Data Serving Capabilities*, SG24-8518 points out that Linux applications benefit highly from the highest level of availability in the industry. Starting with the z16 generation an availability of 99.9999999% was achieved.[2]

This is done both on the hardware level, with redundant hardware components and redundant processor execution steps and integrity checking[3], and enterprise-ready firmware and middleware.

AI solutions, such as AI-powered fraud detection, deliver a tremendous value to financial institutes and requires a high level of reliability and availability. Along with this technical consideration, AI solutions on IBM Z and LinuxONE also prioritize content reliability and availability, trustworthy AI and AI governance, enabling scalable, business-critical, enterprise ready AI applications.

3.1.2 Secure

As business-critical data and data protected under data protection laws are used for AI solutions in an enterprise ready AI application, the data needs to not only be stored secure at the storage site (Data-at-Rest Encryption), but also during the transfer of network (Data-in-Transit Encryption) and during the workload period (add).

With IBM Secure Execution for Linux, which enables a Trusted Execution Environments, helping ensure that applications and data are securely separated, minimizing risks and enhancing security.

The support of quantum-safe cryptography prepares for future post-quantum times, in which quantum computers are being used to encrypt data.[4]

With this wide range of capabilities, IBM LinuxONE and Linux on IBM Z enables organizations to implement Confidential Computing.

[2] https://newsroom.ibm.com/2024-02-06-New-IBM-LinuxONE-4-Express-to-Offer-Cost-Savings-and-Client-Value-through-a-Cyber-Resilient-Hybrid-Cloud-and-AI-Platform

[3] https://www.redbooks.ibm.com/redbooks/pdfs/sg248518.pdf, p. 46

[4] https://www.ibm.com/linuxone/security

Note: IBM z17 is designed to

- Process up to 35 billion encrypted requests per day with OLTP applications.
- Run OpenSSL bulk encryption with AES-256-XTS with up to 1.6x more throughput versus on a compared x86 system.
- Scale up your I/O intensive Linux application and protect your data at rest with up to 15 million read-only I/O operations per second and 7.5 million read-write operations per second to an encrypted block device with FCP attached storage.

3.1.3 Scalable

IBM LinuxONE and IBM Z uses an infrastructure, such as IBM Telum and II, enabling vertical and horizontal scaling. As software licensing is a major cost block, the higher processor utilization results in lower software licensing, when licensing per core is the measurement.

For more information on Achieving Sustainability, Security and Scalability with IBM Z and LInuxONE 4, see https://www.ibm.com/downloads/documents/us-en/10a99803f5afd8cf

Note: IBM z17 is designed to :

- Run an AI infused online transaction processing (OLTP) workload on OpenShift Container Platform requiring up to 5.6x fewer cores versus running it on a compared x86 platform.
- Run Red Hat Enterprise Linux with KVM, deploying up to 3,000,000 NGINX containers.

3.1.4 AI acceleration

AI acceleration refers to the use of specialized hardware, software, or architectural optimizations to significantly speed up the processing of artificial intelligence (AI) workloads, such as machine learning (ML) model training, inference, or data preprocessing. It aims to enhance the performance, efficiency, and scalability of AI applications, particularly in enterprise environments where reliability, security, and scalability are critical. AI accelerators are critical to processing the large amounts of data needed to run AI applications.

IBM Z and LinuxONE hardware architectures are designed with dedicated and highly energy efficient AI acceleration in the form of on-chip acceleration through the use of Telum and II and PCIe acceleration in the form of Spyre.

For more information on Telum II and Spyre, see https://www.ibm.com/new/announcements/telum-ii

3.1.5 Available AI frameworks and tools for Linux

As listed in Chapter 2, "Optimized Model Serving Solutions" on page 9, IBM delivers a wide variety of products for building and implementing enterprise ready AI applications, supplemented by a wide variety of ISV software and open-source software.

Note: For more insights on distribution channels and how to implement the available resources in your own applications, see:
https://ibm.github.io/ai-on-z-101/

You will also find assets and guidance on how to best use the Linux s390x architecture.

Figure 3-1 provides a subset of the AI ecosystem's available AI frameworks for IBM Z and LinuxONE.

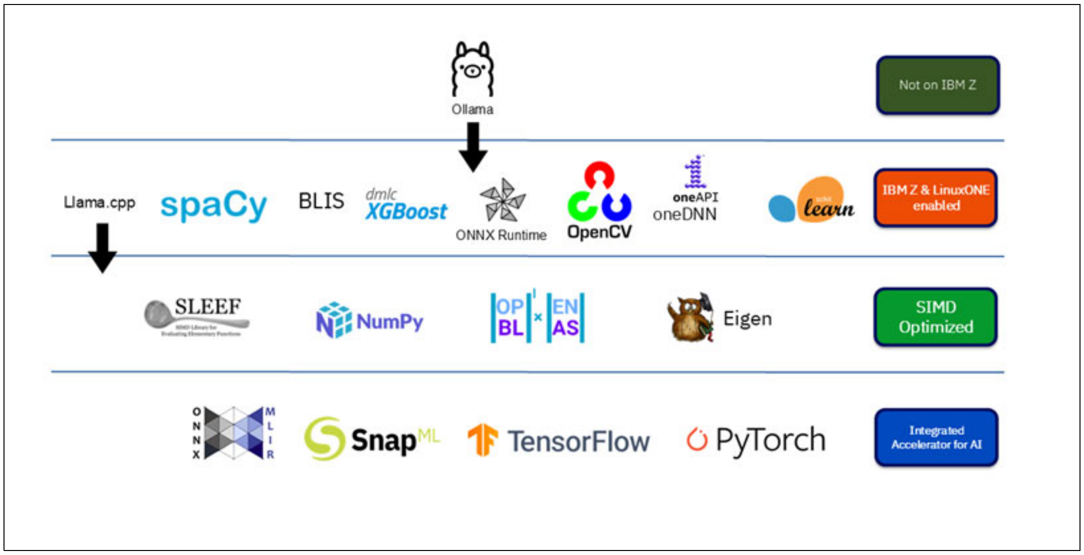


Figure 3-1 Subset of available AI frameworks for IBM Z and LinuxONE

The IBM z17 has been designed to optimize open-source AI frameworks to exploit the SIMD instruction set, Integrated Accelerator for AI in IBM Telum and II and the latest AI Accelerator IBM Spyre into the IBM Z and LinuxONE architecture, called s390x[5].

In adherence with the high secure standards of the IBM Z and LinuxONE infrastructure, IBM is providing these AI frameworks with additional security checks and distributes the frameworks via their own registry.

See chapter 2 in Obtaining open source AI packages on IBM Z and LinuxONE for more information on obtaining open source AI packages with optional IBM support.

3.2 The AI lifecycle methodology

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 3-2.

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, or fine tune an existing LLMs with your own data 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

Figure 3-2 AI lifecycle methodology

5 https://www.redbooks.ibm.com/redpieces/pdfs/sg248579.pdf

For details on developing and deploying an ML model on IBM z/OS, Linux on IBM Z or LinuxONE and using an AI lifecycle methodology, see Chapter 2 of the IBM Redbooks publication, *Optimized Inferencing and Integration with AI on IBM zSystems: Introduction, Methodology, and Use Cases*, REDP-5661.

Getting started with AI can take some planning and time. IBM offers a no-charge discovery workshop that is designed to help you enable AI on IBM Z and LinuxONE technologies and provide project planning and implementation guidance.

The following are the basic steps of the process that can be used to enable AI in your enterprise with an AI Discovery Workshop.

1. AI on IBM Z and LinuxONE Discovery Workshop
2. Proof of Concept which utilizes solutioin templates
3. Production planning

In the following sections we will focus on the business perspective and best practices to successfully deploy an enterprise AI solution end-to-end, starting with involved personas and designing a project plan. This project plan can be used and adapted by your organization to deliver a fully production-ready AI solution on LinuxONE or Linux on IBM Z. The chapter will be concluded by an anonymized end-to-end client implementation, which can be used as an example.

3.2.1 Starting the Proof of Concept: Personas

A Proof of Concept (PoC), in which you test a product in a given timeframe, (typically three months), should be conducted to validate the feasibility of your solution before committing to full-scale development or investment. It can serve as a low-risk, focused experiment to test critical assumptions and reduce uncertainty.

For deploying an enterprise AI solution, defining the business problem that needs to be solved is key. Delivering the business value of an AI solution in the most optimal, secure, reliable and high-performance way is the Key Performance Indicator (KPI) of any AI project. The business goals should be documented and understood by all stakeholders.

Gather key personas such as the stakeholder(s) and project team lead(s) to clearly define the objectives and scope:

- Clearly define your objectives - what do you want to achieve, what business purpose will this serve. Examples of some objectives could be the following:
  - A recommendation system
  - Automate repetitive tasks
  - Analyze data for faster decision making (identify fraud, examine insurance claims, anti-money laundering etc)
  - Personalize customer experience (use a Chatbot, etc)
  - Faster forecasting
  - Optimize processes to cut costs
  - Confidential (Secure Execution)
- Clearly define the purpose of the PoC (for example, test a specific feature, technology, or process).
- Identify key success criteria and measurable outcomes.
- Create a document of understanding (DOU) clearly outlining the objectives, scope of the PoC and the roles and responsibilities of the personas that will be involved in the project.

Once you have clearly defined the purpose and success criteria of the PoC with key stakeholders, you will involve additional team members.

Often the teams that are responsible for operating the enterprise AI solutions (infrastructure personas) are lacking knowledge of the business challenges, but do have the infrastructure knowledge and know the unique components of their infrastructures.

Application developers, data architects or data scientist (business personas) work daily on solving business challenges, but often lack infrastructure knowledge.

Combining the infrastructure personas (IT, DevOps, data engineers) with the business personas (domain experts, product managers, stakeholders) in an AI project is critical for ensuring a project's success, alignment with organizational goals, and effective deployment of enterprise-ready AI on IBM Z and LinuxONE solutions. This collaboration bridges technical and business perspectives, enhancing reliability, scalability, security, and cost-efficiency (TCO).

3.2.2 Project planning for a Proof of Concept and production implementation

Understanding the business challenges and bringing together the infrastructure and business personas are key for any project and this includes a successful AI on IBM Z or LinuxONE project.

Equally important is documenting the key elements of the project plan to understand the steps needed to bring the PoC to "life" and serve as a basis for production implementation.

Table 3-1 provides a sample of some of the tasks required in planning, which would be included in the DoU.

Table 3-1 Sample task list for PoC

Step	Task	Planned duration / Date	Status	Responsibility
	Meet with key Personas	Pre-PoC	Done	Stakeholder Project lead
	Project setup and staffing	Pre-PoC	Done	All
	Fulfill hardware and software requirements (see installation roadmap)	Pre-PoC	Done	Infrastrucure team
	Provide a test license for the product	PoC	Done	Product company
	Download and installation of the product	PoC	Done	Infrastructure team
	Documentation and evaluation of results	PoC	Open	All
	Final PoC outcome presentation	PoC	Open	All

After the PoC has completed and the final outcome is acceptable, production implementation can begin as a new project.

Concentrate on this PoC to evaluate optimal integration with your existing infrastructure and identify vulnerabilities or gaps in the enterprise-ready AI solution and modify your production task list appropriately.

3.2.3 Example implementation of a Proof of Concept

This chapter describes an engagement with a client, named client Y, which can be used as a blueprint for your implementation of an enterprise ready AI application at scale.

Client Y's application department sought to implement AI-enhanced fraud detection. Previous engagements showed that integrating AI models with their traditional rule-based approach could yield better results. Recognizing the need to address business challenges like rising fraud, the department decided to attend an IBM conference. They were aware that financial transactions were processed through IBM Z but needed more information.

After recognizing the benefits of implementing AI on IBM Z and LinuxONE, and understanding that only this infrastructure could provide the necessary reliability, security, and scalability for enterprise-scale AI applications like AI-enhanced fraud detection, the application department decided to proceed. They included both IBM and the infrastructure department in their plans. An AI Discovery Workshop was deemed unnecessary, as the business challenge was clear and no further education on AI or the IBM Z and LinuxONE architecture was required.

The decision was made to validate the feasibility of a production implementation through two Proofs of Concept (PoCs), further described in Figure 3-2.

- The business PoC focused on developing an appropriate AI model for AI-enhanced fraud detection.
- The technical PoC concentrated on the technical aspects of scaling enterprise-ready AI applications, specifically evaluating whether a given software solution on IBM Z could be utilized and integrated into their infrastructure environment.

Table 3-2 Description of two PoCs

Business PoC	Technical PoC
► Train and evaluate AI ► Choose features ► Load tests → Can it be developed?	► Installation of product ► Test of scalability and integration into existing infrastructure ► Measurement of resource requirements → Can it be operationalized?

This method allows for targeted focus on the specific questions that must be addressed before making a decision. The infrastructure and application team at Client Y working together ensured ongoing collaboration and synchronization to support the implementation of an AI-infused fraud detection solution.

After demonstrating the effectiveness of the AI-powered fraud detection solution on IBM Z, Client Y chose to proceed and deployed the solution at scale in collaboration with IBM.

The process followed by this client for their PoCs can be summarized in the following way:

- The application department identified their business challenge
- The application department engaged their infrastructure department and IBM
- IBM was engaged to address challenges and answer questions early
- A DOU was created along with a project plan
- The team had regular interlock meetings throughout the life of the project

3.3 Examples of use cases

This section provides some inspirations for use cases not only in the area of traditional AI (predictive AI), but also by the inclusion of Generative AI, which enables a cross-engagement, called multiple AI model architecture.

For further insights on a technical implementation of a real-time, in-transaction scoring use case, we recommend Chapter 3 of the IBM Redbooks publication, *Optimized Inferencing and Integration with AI on IBM zSystems: Introduction, Methodology, and Use Cases*, REDP-5661. Chapter 4 of this same IBM Redbooks publication describes predictive AI use cases in addition to the predictive AI use cases in this publication.

Figure 3-3 shows some of the AI on IBM Z and LinuxONE use cases that are enhanced by the improved capabilities of Telum II and Spyre.

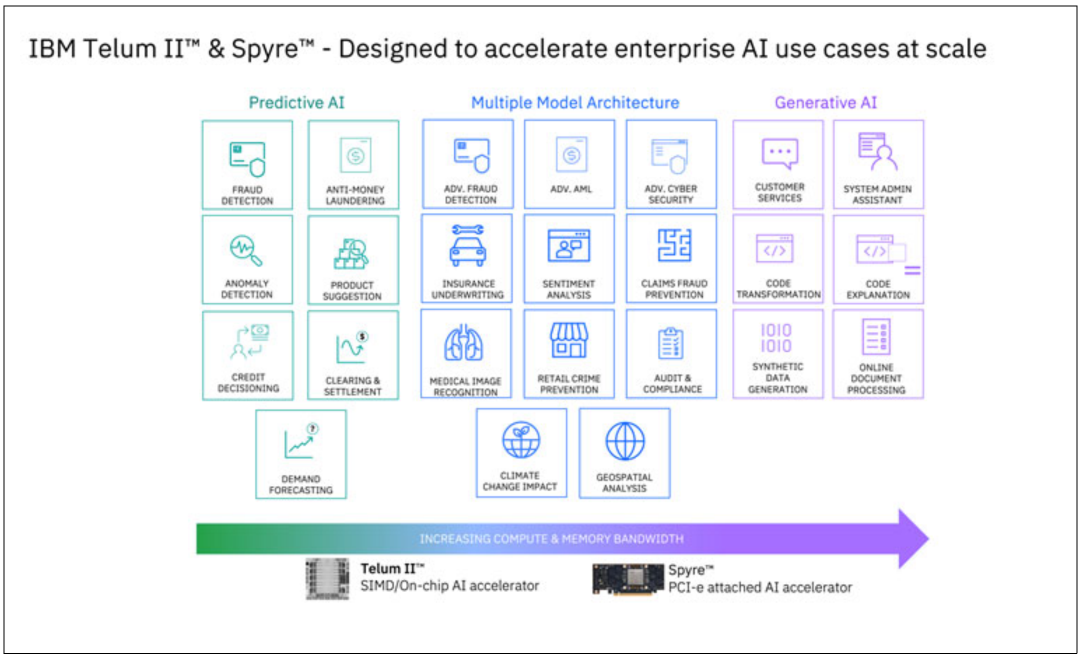


Figure 3-3 Current AI on IBM Z and LinuxONE use case landscape

> **Note:** Not all use cases are shown in Figure 3-3. For more information on an AI on IBM Z and LinuxONE Discovery Workshop, contact your IBM Representative or the AI on IBM Z and LinuxONE team directly at aionz@us.ibm.com

3.3.1 Predictive AI

Predictive AI refers to AI systems that are designed to analyze historical data and make predictions about future events or trends by leveraging Machine Learning and Deep Learning models. For more information, see Chapter 1 of the IBM Redbooks publication, *Optimized Inferencing and Integration with AI on IBM zSystems: Introduction, Methodology, and Use Cases*, REDP-5661

Predictive AI models, being robust for decision-making and prediction while lightweight, are ideal for use cases involving real-time decision-making and prediction within AI-infused transactions.

In time-critical transactions where milliseconds are crucial, external API calls are inadequate for meeting Service Level Agreements. The Telum and Telum II processors, equipped with an AI acceleration unit per chip, are designed to be highly optimized for such workloads, making them well-suited for use cases requiring rapid, embedded AI processing.

- low-latency
- a secure environment due to sensitive data
- no external API call, since the transaction is already run on IBM Z or LinuxONE

Predictive AI is well-suited for use cases requiring Trustworthy AI, as mandated by legal frameworks such as the EU AI Act. While Trustworthy AI is achievable for Generative AI, it is generally easier to implement with Predictive AI models.

Note: IBM Z products have won several awards in the area of Trustworthy AI.[a] watsonx.governance is offering dedicated trustworthy AI functionalities for generative AI.[b] AI Factsheets and OpenScale within IBM Cloud Pak for Data are offerings for trustworthy AI for predictive AI models, available for Linux, as described in Chapter 2, “Optimized Model Serving Solutions” on page 9.

a. For the 2024 award of the Business Intelligence Group for Machine Learning for IBM z/OS, please look here: https://community.ibm.com/community/user/ibmz-and-linuxone/blogs/abid-alam/2024/03/20/mlz-ai-excellence-award?communityKey=038560b2-e962-4500-b0b5-e3745175a065
For the 2025 award of the iF Design Award for Machine Learning for IBM z/OS, please look here: https://ifdesign.com/en/winner-ranking/project/machine-learning-for-ibm-zos/687833
b. https://www.ibm.com/products/watsonx-governance

3.3.2 Generative AI

Generative AI refers to a type of artificial intelligence that can create new content, such as text, images, music, or even code, based on the data it has been trained on. Generative AI can be used for advanced AI applications.

The following are some examples of using generative AI.

- System administrative assistant

  When the infrastructure department faces an increased workload but lacks sufficient personnel, an assistant can be invaluable. IBM's watsonx Assistant for Z addresses this need by providing a dedicated product that offers the option to complete system administration tasks directly from the chat interface.

  Note: For more information on how IBM's watsonx Assistant for Z can help you, see: https://www.ibm.com/products/watsonx-assistant-for-z

- Code explanation and optional code transformation

  To address the growing need for understanding and modernizing legacy code, IBM offers a specialized product for mainframe infrastructure called IBM watsonx Code Assistant for Z. This product includes an optional feature for transforming code into Java.

> **Note:** For more information on IBM watsonx Code Assistant for Z, see:
> https://www.ibm.com/products/watsonx-code-assistant-z

- Synthetic data generation

  High-quality data is essential for effective testing and creation of AI models. IBM provides a dedicated solution for this need with IBM Synthetic Data Sets.

> **Note:** For more information on IBM Synthetic Data Sets, see:
> https://www.ibm.com/products/synthetic-data-sets

3.3.3 AI multiple model architecture

Due to their lightweight design, Predictive AI models excel in execution speed but lack depth in processing unstructured data or providing detailed decision explanations. Generative AI can complement Predictive AI in these areas. multiple AI model architecture approaches can be employed when greater detail is required as long as:

- the transaction is not time critical
- the transaction is time critical, but further exploration or explainability can be done after the transaction.

The following are two industry examples of AI multiple model architectures:

- Insurance industry

  Consider an enterprise-ready AI model for processing hail damage payment claims. A Predictive AI model can efficiently handle structured data, such as payment amounts or no-claim bonuses, to make payment decisions. However, it may benefit from unstructured data, like appraiser photos or police reports, which it cannot process directly. A Generative AI model can analyze this unstructured data, convert it into structured data, and feed it to the Predictive AI model for enhanced decision-making.

- Financial Industry

  In a fraud detection example, Predictive AI models are essential for low-latency decision-making, such as approving or declining payments. However, Generative AI can provide additional value post-transaction by generating further insights or explanations.

  Detailed and automated explanations for declined transactions can be autonomously sent to clients.

3.3.4 AI Security

In the rapidly evolving landscape of artificial intelligence, ensuring robust security measures is paramount. IBM's AI Security initiatives focus on safeguarding AI systems and data through innovative technologies and practices. This section describes three such initiatives of AI security: IBM Secure Execution, which provides enhanced protection for sensitive workloads; Confidential AI, which describes IBM's initiatives to ensure privacy and security in AI processes; and IBM Synthetic Datasets, which offer secure and reliable data for training AI models. Together, these subsections highlight IBM's commitment to advancing AI security and maintaining trust in AI-driven solutions.

IBM Secure Execution

Enterprise-scale AI applications with stringent data security requirements can greatly benefit from IBM Secure Execution for Linux.

IBM Secure Execution for Linux is a security technology introduced with IBM z15® and LinuxONE III. It is designed to protect the data of workloads running in a KVM guest from being inspected or modified by the server environment. This means that no hardware administrator, KVM code, or KVM administrator can access the data within a guest that is started in secure-execution mode.

An example of IBM Secure execution from the healthcare industry considers the capture or X-Rays. Highly sensitive data, such as X-Rays, require secure storage and execution can be protected within a secure enclave. IBM Z and LinuxONE provide the necessary infrastructure for this level of security.

For more information on IBM Secure Execution, see
https://www.ibm.com/docs/en/linux-on-systems?topic=execution-introduction

Confidential and secure AI

As organizations leverage predictive and generative AI, they face challenges in ensuring that these innovations are deployed without misconfigurations, security vulnerabilities or other threats that could led to data protection risks or compliance violations. This is where IBM's approach to confidential AI can help organizations meet their secure AI needs to ensure privacy and security of AI processes and data. Following IBM’s framework to secure generative AI initiatives, organizations should consider approaches that allow them to adequately secure their dataset, model, and utilization of trained and deployed models. Additionally, organizations should evaluate approaches that secure AI model infrastructure and establish proper AI governance.

IBM’s confidential and secure AI strategy leverages confidential computing technologies to protect data during processing by using hardware-based enclaves, which keep sensitive information encrypted and isolated. By leveraging computing technologies for secure AI, organizations can stay protected against threats and attacks on sensitive data and AI models with trust. Through secure AI, organizations can protect their sensitive data by training with artificial data, monitor models for fairness and bias and drive transparency for those models with governance reports.

For more information on the IBM framework for securing generative AI, see
https://www.ibm.com/products/tutorials/ibm-framework-for-securing-generative-ai

IBM Synthetic Data Sets

IBM Synthetic Data Sets is a family of artificially generated, enterprise-grade datasets that enhance predictive artificial intelligence (AI) model training and large language models (LLMs) to benefit IBM Z and IBM LinuxONE clients, ecosystems, and independent software vendors. These pre-built datasets are downloadable and packaged as comma-separated values (CSVs) and data definition language (DDL) files, making them familiar to use, and compatible with everything from databases to spreadsheets to hardware platforms to standard AI tools. These datasets also leverage the IBM industry expertise and domain knowledge of the financial services sector without using any real client seed data, which alleviates security concerns with Personally Identifiable Information (PII).

IBM Synthetic Data Sets trains and enhances predictive models and composite AI methods. Those models can be deployed to IBM Z and LinuxONE with inferencing tools, such as IBM

Machine Learning for IBM z/OS, AI Toolkit for IBM Z and IBM LinuxONE, and IBM Cloud Pak for Data on IBM Z.

The IBM Synthetic Data Sets family contains the following features:

- IBM Synthetic Data Sets for Payment Cards
- IBM Synthetic Data Sets for Core Banking and Money Laundering
- IBM Synthetic Data Sets for Homeowners Insurance

For more information on IBM Synthetic Data Sets, see *IBM Synthetic Data Sets*, REDP-5748

4

Bringing it all together with the use cases

This chapter describes two solutions that are well-suited for IBM Z and LinuxONE. The approach and solution architecture outlined here can be applied to any industry or sector.

Since data is crucial for building and training AI models, the chapter begins with an introduction to several IBM Z and LinuxONE frameworks that accelerate application access to data. The first use case demonstrates how to embed these data solutions within your AI framework.

The first solution scenario involves a multiple AI model approach for detecting fraud in home insurance claims. This scenario uses replicated Db2 z/OS transactional data, accessed via the IBM Data Gate protocol, and augments it with contextual embedding features derived from LLMs. The merged data is then used as input for a neural network framework to detect potential fraud. This multiple AI model approach can be applied to other industries, allowing clients to benefit from combining predictive and Generative AI for higher precision and accuracy.

4.1 Use case: An advanced multiple AI model framework for home insurance fraud detection processing

In this section, we describe the pattern of an advanced multiple model AI framework that combines traditional and generative AI approaches to detect potential fraud in high-volume home insurance claims with high precision and accuracy.

This integrated approach enables insurers to leverage generative AI technology to process unstructured claims narrative data, deriving meaningful context embeddings that augment structured insurance claims data.

IBM Z, equipped with a Spyre AI Accelerator, is designed to provide an ideal AI infrastructure for implementing this combined model. This architectural approach, which integrates generative AI with predictive AI, can be applied across various industries and domains, not just within the insurance sector. The solution includes the Data Gate technology described in this chapter, which serves as a mechanism to replicate structured Db2 for z/OS table data into the lakehouse.

The Business impact of deploying the entire solution on IBM Z:

- Reduced insurance claims lifecycle
- Improved customer satisfaction via prioritizing the claims as advised by the framework
- Improved insurance claims team efficiency
- High confidence of the processing and scoring of 100% transactions at scale.

4.1.1 Overview, challenges and needs for fraud detection in insurance

According to a study from Celent, fraud generated an estimated US$385 billion globally in losses to the banking, cards and payments sectors in 2021. The insurance sector suffered from losses associated with fraud as well. PwC has estimated these losses to be US$308.6 billion in the US alone[1].

As financial crime mechanisms become increasingly complex and sophisticated, enterprises must adopt more advanced methods for detection, adapting to modern challenges and enhancing their fraud detection capabilities. While traditional predictive AI algorithms have demonstrated a certain level of accuracy, complementing them with generative AI (GenAI) frameworks opens new opportunities for improved fraud detection and prevention.

A more modern approach would be an advanced multiple AI model system that leverages both predictive AI and GenAI for enhanced insurance claims fraud detection. Deploying this solution on-premises on IBM Z can address several industry challenges, including guaranteed SLAs for insurance transaction processing, secure execution of sensitive data, and compliance with industry standards. IBM Z provides an optimal infrastructure for dynamically scaling resources in a sustainable manner. The Telum II and Spyre Accelerator on IBM Z contribute to fast and secure AI inferencing.

4.1.2 Advanced multiple AI model insurance claims fraud detection: A reference Architecture on IBM Z or LinuxONE

This architecture contains the following main components:

[1] https://www.pwc.com/gx/en/industries/financial-services/publications/financial-crime-in-the-insurance-industry.html

- Db2 z/OS
- Lakehouse
- Data Gate
- Watsonx.AI
- Triton Inference Server

Data is at the heart of this process, and insurance claims processing involves several steps. Let's explore the collection and curation of data, followed by data merging for enhanced AI model development and training (see Figure 4.2). A Db2 for z/OS table with structured claims data is replicated to an IBM watsonx.data® lakehouse through the high-speed Data Gate integrated synchronization protocol and then written to the Apache Iceberg open table format for further processing. The watsonx.data lakehouse merges the structured claims data from Db2 with context embedding content extracted via a large language model for AI model development and training.

The Data Gate protocol automatically detects updates made to the source table and replicates the changes into the lakehouse, ensuring the latest data replica. Next, features, such as claim descriptions and narratives, are extracted from the unstructured claims narrative using a large language model running on IBM Z, which can be accelerated via IBM Z Spyre cards. Extracted features are converted to BERT encoder embeddings inferred sentiments which allow to capture the nuances and specifics of each claim, focusing on the urgency statement derived from the claim description.

These extracted features will be used for downstream tasks such as classification or regression to adequately rank claims. Predictive AI model scoring will identify high risk patterns and flag claims that exhibit characteristics of potential fraud. The model is then deployed using Triton Inference Server as it enables for efficient, scalable AI inferencing allowing to quickly and identify fraudulent home insurance claims, and help organizations prioritize attention for optimal customer service and resource utilization. Overall, by leveraging large language models for deep textual understanding and neural networks for pattern recognition in structured data, the multiple AI model approach significantly improves the accuracy of fraud detection. IBM Z hardware acceleration enhances overall performance and scalability.

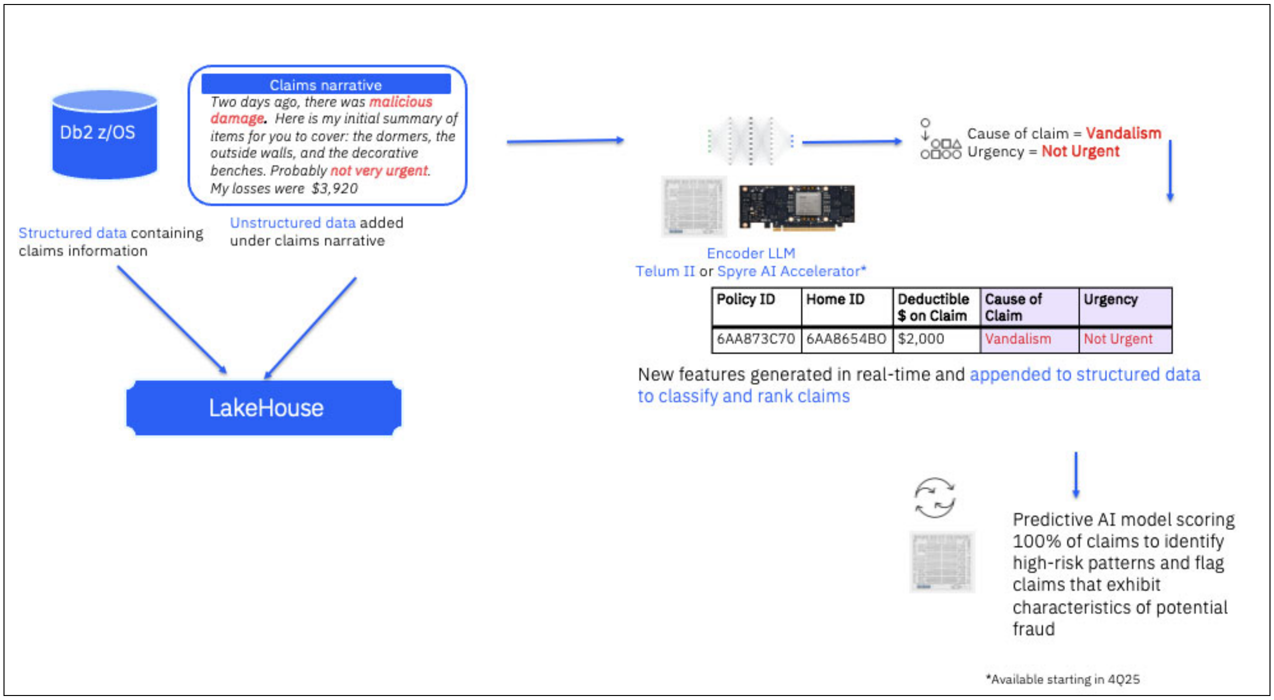


Figure 4-1 Advanced Multi Model insurance claims fraud detection on IBM Z

4.2 Use case: An advanced multiple AI model framework for anti-money laundering

In this section, we describe the pattern for an advanced real-time multiple AI model framework that combines traditional and generative AI approaches to prevent money laundering with high precision and accuracy, enhancing illicit activity screening. This integrated multiple AI model approach enables financial institutions to leverage generative AI technology to augment traditional predictive AI with real-time Know-Your-Customer verification enrichment through large language model (LLM) processing of unstructured data, including social media, news, online forums, and other sources.

Real-time multiple AI model inferencing on IBM Z improves overall system accuracy, allowing both financial institutions and regulators to achieve higher accuracy and gain new business insights from all available data. IBM Z, equipped with a Spyre AI Accelerator, provides an ideal AI infrastructure for implementing this combined model. This architectural approach, which integrates generative AI with predictive AI, can be applied across various industries and domains, not just within the financial sector.

The business impact of deploying an entire solution on IBM Z:

- Detection of illicit activity
- Responding in real-time to save money on regulatory compliance
- Safeguard economic transparency.

4.2.1 Anti-Money Laundering Implementation: Overview, Challenges, and Requirements

As money laundering becomes increasingly sophisticated, prevention and response requires a robust AI-driven approach. According to a Celent study, regulators frequently heavily fine banks for inadequate anti-money laundering (AML) programs. Many AML operations suffer from high false positive rates, imposing a severe burden on many financial institutions.

When money laundering is left unchecked, it enables insidious activities such as drug trafficking, human smuggling, and even terrorism on a global scale.

Reviewing a modern approach could show an advanced multiple AI model system that uses the power of both predicative AI and GenAI for AML detection and verification. Deploying this solution on-premises on IBM Z solves the following industry challenges: a guaranteed SLA for the financial transaction processing; secure execution of SPI and other sensitive data; compliance with industry standards.

Deployment on IBM Z provides an optimal infrastructure for dynamically scaling the resources in the sustainable way. Telum II and Spyre Accelerator on IBM Z contribute to the fast and secure AI inferencing.

4.2.2 Reference architecture of advanced Multi Model AI AML framework on LinuxONE

Anti-money laundering (AML) screening is inherently complex and involves multiple steps. An internal IBM benchmark demonstrated that a cumulative, composite approach using both predictive and generative AI (GenAI) models yields a higher degree of precision and accuracy in detecting illicit activities. While standard AML processing includes AI transaction monitoring and AI transaction route monitoring, LLM-infused AML can leverage a variety of unstructured data sources, such as existing KYC data, social media activity and inactivity, online forums, and more.

This innovative approach enhances accuracy and provides new insights based on all available data. The AI on Z toolkit, which combines containerized open-source inferencing platforms, offers an inferencing layer to support AML implementation on LinuxONE.

Let's review an advanced multiple AI model framework for screening financial transactions to prevent money laundering. In Figure 4-2, the first step shows transaction patterns that are identified based on historical payment data using graph algorithms. As a result of AI model inference, hidden relationships and interactions are compared in real-time against common (and uncommon) AML patterns.

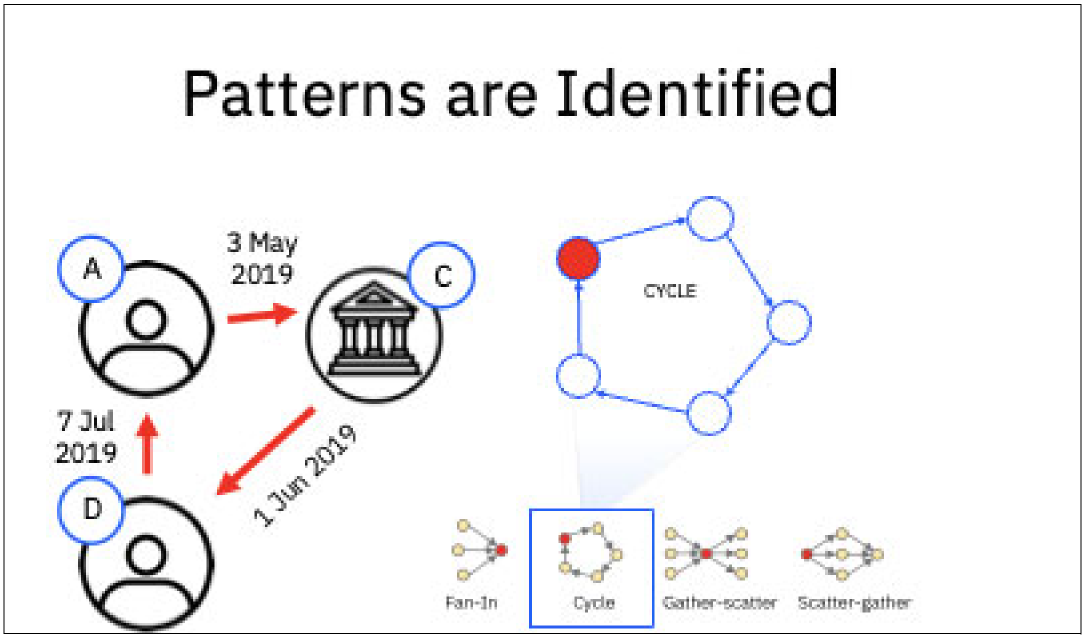


Figure 4-2 A process of identifying patterns and relationships of financial transactions

Next, the AI model takes all key features into account on a weighted basis - like suspicious transaction amounts, the number of transactions in each cycle, and other anomalies compared with prior activity (Figure 4-3).

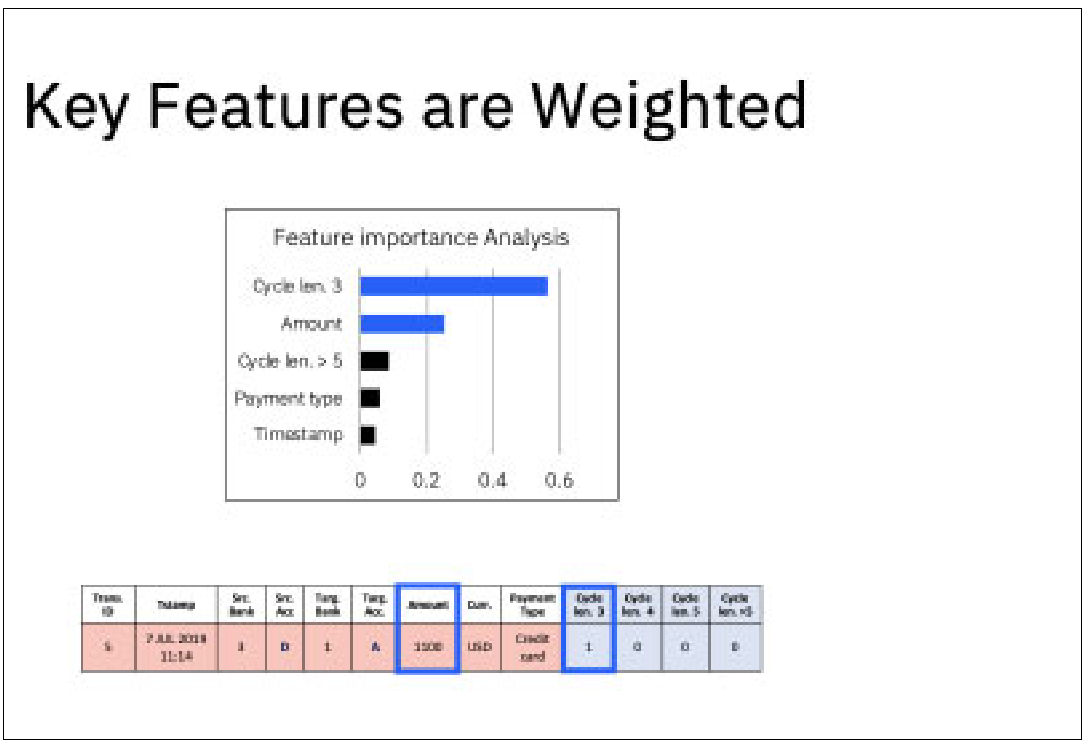


Figure 4-3 Key features are weighted

In the next step (Figure 4-4), we enhance the overall framework by infusing the AML screening process with GenAI sentiment analysis, performed using large language models (LLMs). This involves processing vast amounts of unstructured data, such as social media and KYC information, to uncover previously hidden data and relationships that may indicate illicit transactions.

Figure 4-4 Unstructured data processing

Finally, considering a wide array of variables and all data and insights obtained on previous steps, and becoming more refined with each new transaction, the AI model reaches an accurate, explainable recommendation to determine the likelihood of being an illicit activity.

Infusing this framework with LLM sentiment analysis can significantly enhance overall accuracy. Further fine-tuning of the LLM could potentially improve accuracy even more.

Leveraging LLM, the system is making logic-driven predictions on unstructured data, enhanced illicit screening and increasing model accuracy.

4.3 Use case: Credit risk assessment - Triton Inference Server (TIS) and Open-Source Framework

In this use case, we discuss leveraging AI to enhance credit risk assessment for financial institutions. This use case integrates data management, machine learning, and real-time decision-making to improve operational efficiency, fraud prevention, and customer experience.

Some of the key components include data management. Our example utilizes IBM Db2 for z/OS data integration and management.

An AI-driven assessment employs IBM Machine Learning and SnapML for model training and deployment. Real-time decision making provides instant credit approval or rejection based on AI inference.

Analytics dashboards are included for monitoring trends and retraining models for continuous improvements in maintaining accuracy.

End-to-End Flow

In this section, we provide an example of an AI architecture for a credit risk assessment system on IBM LinuxONE. For a credit risk assessment system on IBM LinuxONE, data storage plays a crucial role in providing a high-performance file system that can handle large volumes of data.

The entire end-to-end flow is shown in Figure 4-5.

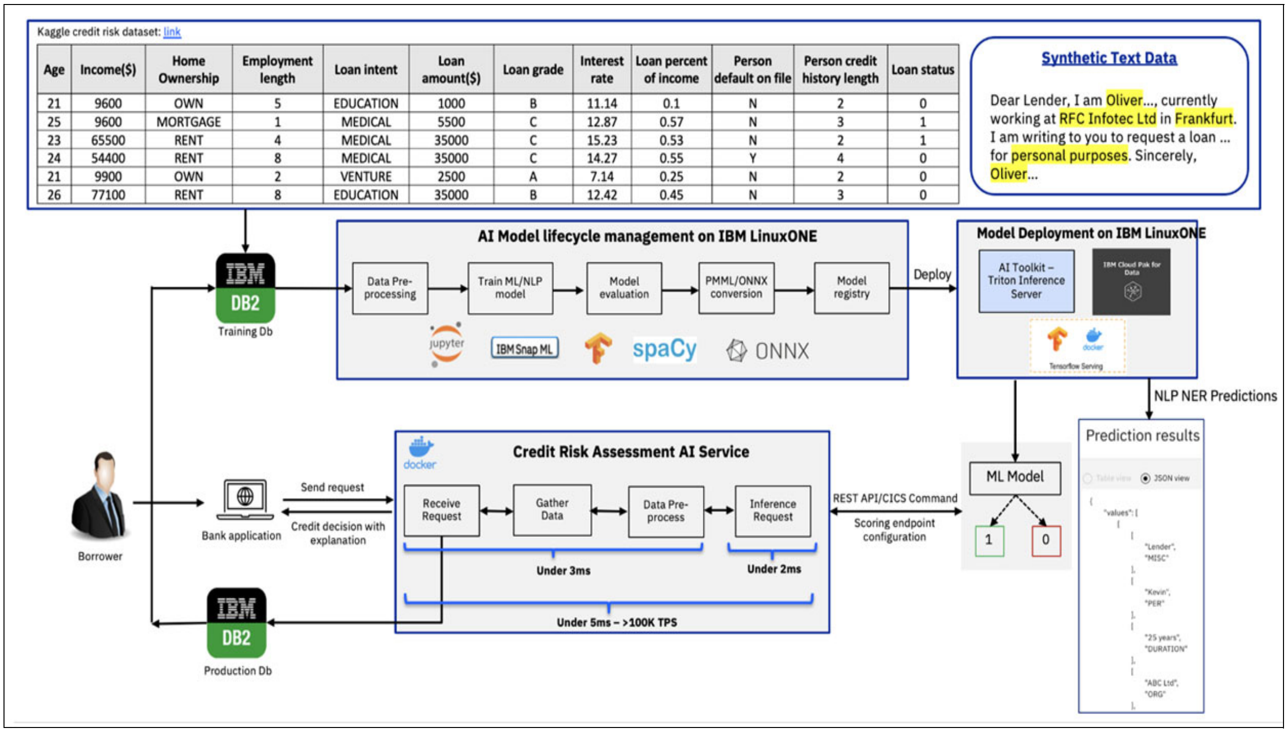
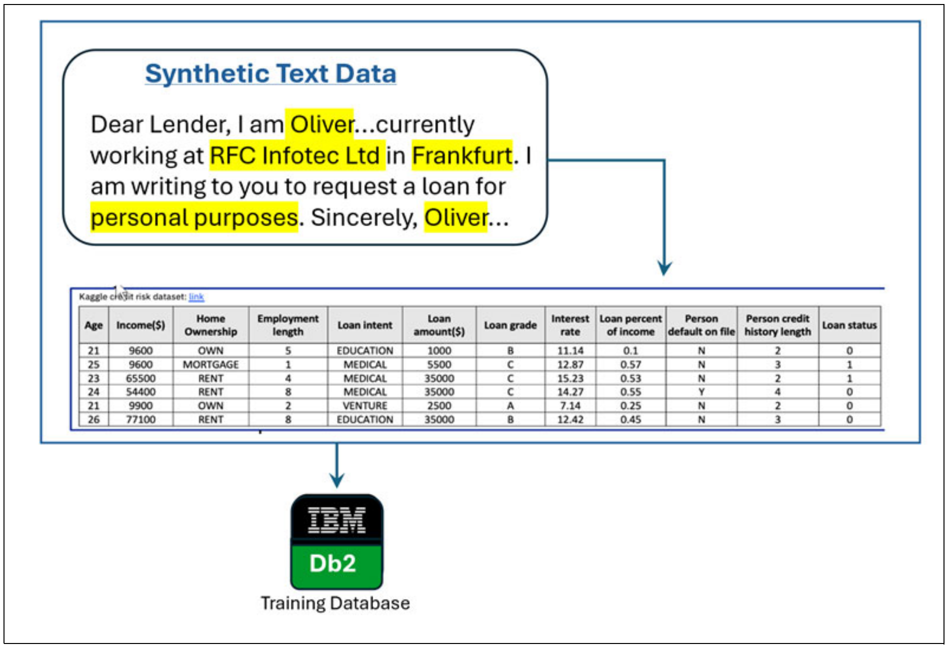


Figure 4-5 AI architecture of credit risk assessment system on IBM LinuxONE

In our example, a sample of synthetic text data (Figure 4-6) is shown which is used to train the AI system. The synthetic data can be used to augment the existing dataset, improving the model's ability to generalize and handle diverse inputs. Since we are using synthetic data to train the AI system, the system would be trained on historical credit data, not personal data.

IBM Synthetic Datasets is available to be used as your synthetic text data. They are artificially generated datasets created to mimic real-world data while preserving privacy and security. These datasets have been designed for use in testing, machine learning model development, and research, offering realistic patterns and structures without exposing sensitive information.

In this example, the IBM DB2® training database stores the raw dataset for training.



Age	Income($)	Home Ownership	Employment length	Loan intent	Loan amount($)	Loan grade	Interest rate	Loan percent of income	Person default on file	Person credit history length	Loan status
21	9600	OWN	5	EDUCATION	1000	B	11.14	0.1	N	2	0
25	9600	MORTGAGE	1	MEDICAL	5500	C	12.87	0.57	N	3	1
23	65500	RENT	4	MEDICAL	35000	C	15.23	0.53	N	2	1
24	54400	RENT	8	MEDICAL	35000	C	14.27	0.55	Y	4	0
21	9900	OWN	2	VENTURE	2500	A	7.14	0.25	N	2	0
26	77100	RENT	8	EDUCATION	35000	B	12.42	0.45	N	3	0

Figure 4-6 Synthetic text data

As shown in Figure 4-7, the dataset is cleaned and prepared by using a Jupyter Notebook. Machine Learning (ML) and Natural Language Processing (NLP) models are trained by using IBM Snap ML (a library for scalable ML) and spaCy (an NLP library). Snap ML handles structured data (numerical features like income, loan amount), while spaCy processes unstructured text (e.g., synthetic text data).

In model evaluation, the trained model's performance is assessed for accuracy and precision. The model is then converted to Predictive Model Markup Language (PMML) or Open Neural Network Exchange (ONNX) formats for deployment.

The converted model is stored in a registry for versioning and deployment (Figure 4-8).

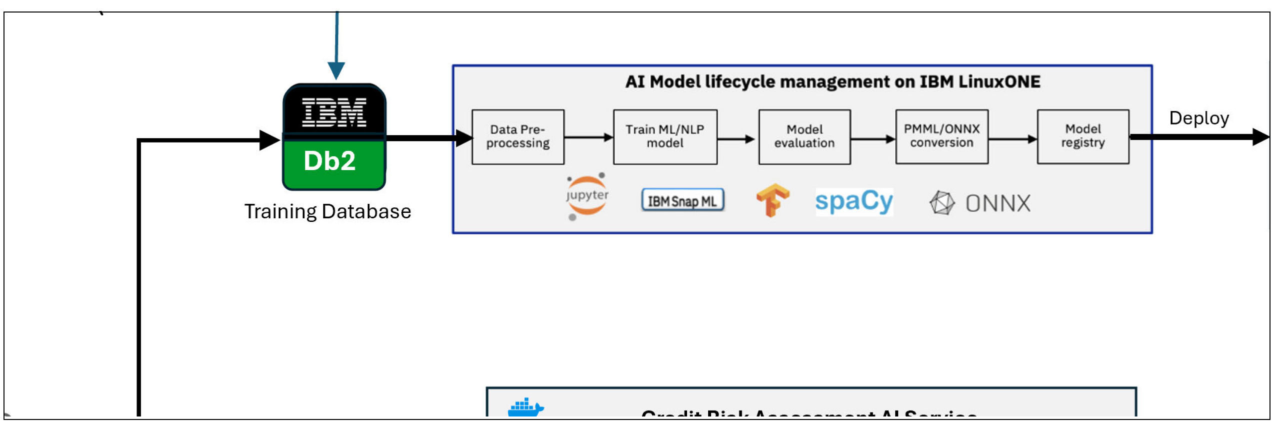


Figure 4-7 AI model lifecycle management on IBM LinuxONE

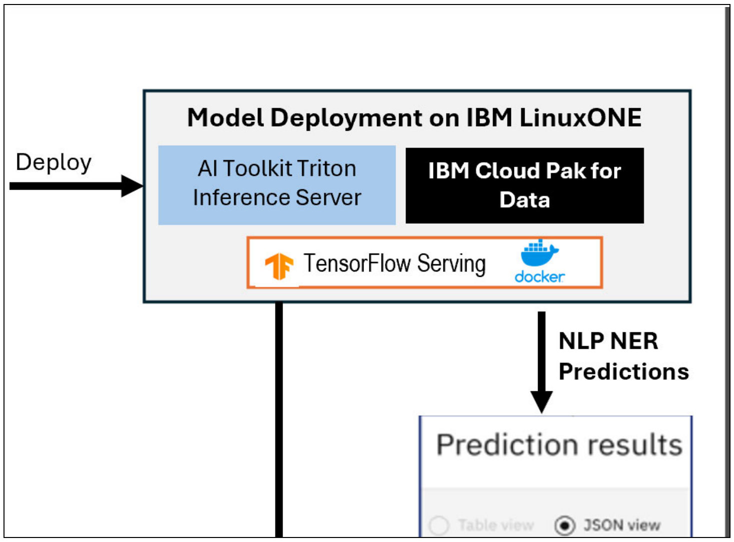


Figure 4-8 Deployment phase of the AI model lifecycle

In the deployment phase of the AI model lifecycle, the trained model is made available for real-time predictions. The AI Toolkit Triton Inference Server is used to serve the trained machine learning (ML) model for credit risk prediction, ensuring fast and scalable inference on IBM LinuxONE.
IBM Cloud Pak for Data is the platform used to manage the AI lifecycle, including data preparation, model training, and deployment. It likely handles model versioning, monitoring, and governance, ensuring the deployed model is reliable and compliant with enterprise standards.

TensorFlow Serving is a system for serving TensorFlow models in production.It works alongside Triton Inference Server, providing flexibility for different model types.

Docker containers are used to package the model and its dependencies, ensuring consistency and portability across environments on IBM LinuxONE. This containerization enables seamless deployment and scaling of the inference services.

Once a borrower makes a request via the bank application, data collection and integration begin, where the borrower details, historical data, and alternative sources are unified. The comprehensive profiling of IBM Db2 for z/OS can ensure data quality and consistency is used. This comprehensive profiling might catch inconsistencies like a borrower with a negative income or a missing loan status, which could skew the AI model's predictions.

Figure 4-9 shows the prediction results of the credit risk assessment system. This system has two types of predictions:

- ML Model Predictions: The ML model outputs a binary classification (1 or 0) indicating credit risk (1 = high risk of default, 0 = low risk).
- NLP NER Predictions: The NLP model (trained with spaCy, as shown in the larger diagram) processes text data and outputs named entities.

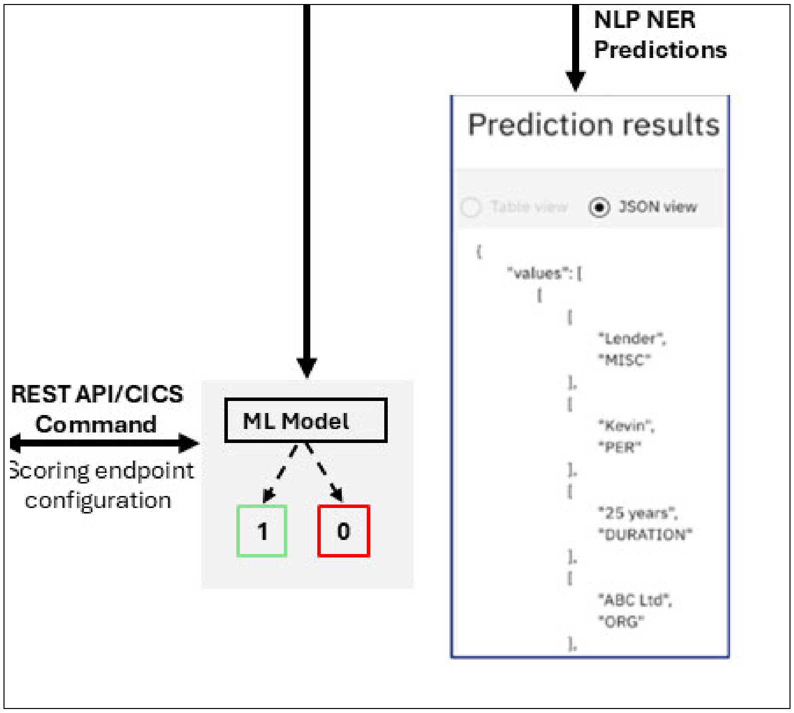


Figure 4-9 Prediction results of the credit risk assessment system

Machine learning models, such as Random Forest and Boosting Classifiers, are trained on this data to predict loan status, achieving high accuracy with metrics like ROC AUC scores. The trained models are deployed on IBM Machine Learning for z/OS, creating scoring endpoints for real-time inference.

Figure 4-10 focuses on how the system processes a borrower's loan application in real-time, delivers a credit risk assessment, and returns a decision to the bank application.

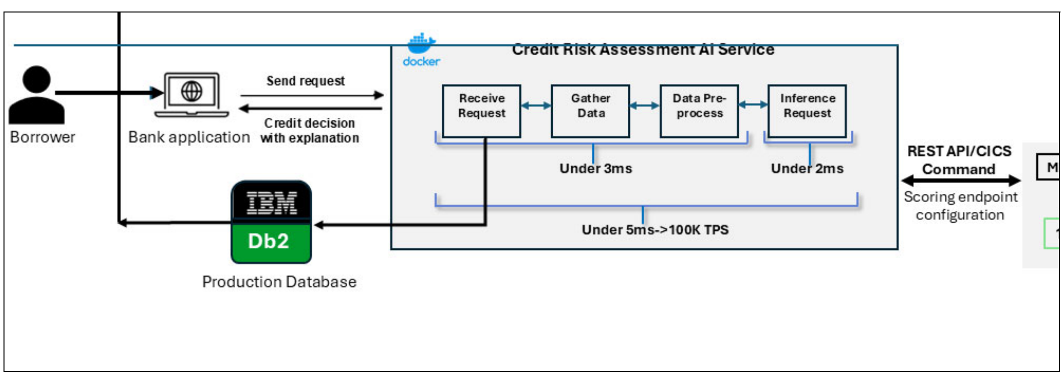


Figure 4-10 Delivering the credit risk assessment

A borrower submits a loan application via a bank application and the system predicts credit approval or rejection with explainability tools enhancing transparency. The application sends a request to the credit risk assessment AI service.

The credit risk assessment AI service receives the borrower's data and pulls relevant data from the production database. Data is then preparedfor inference. The ML model (Figure 4-9) processes the data and predicts the credit risk. A decision (approve/deny) is then returned with an explanation to the bank application.

Continuous learning ensures model accuracy by retraining on production data to handle data drift. This end-to-end process improves operational efficiency, reduces fraud, enhances customer experience, and empowers banks with data-driven decision-making.

4.4 Overview of additional use cases

Chapter 4 has focused on examples of multiple AI model solutions; however, there are other industries and use cases that are also well suited for Linux on IBM Z and LinuxONE technologies. The following are some additional industry use cases where we illustrate their Business scenario, business impact, and example solutions. All run on Linux, including Linux on IBM Z and LinuxONE.

Banking

Know your customer based short loans

Business scenario: A client wants to predict the 'likelihood to repay' for point-of-sale loans to good end-user candidates. This will allow end users to employ their banking relationships as a credit account overdraft.

Business impact: The client can now achieve near real time AI prediction of end user 'likelihood to repay', ena-bling an extended revenue stream for short term loan usage fees.

Example solution:

- Data/Application: DB2 LUW
- AI Platform/Framework: Cloud Pak for Data

Improved Operations for Credit Card Applications

Business scenario: A client wants to move fraud detection models to the IBM Z environment to improve scale, availability and response time issues currently experienced with the current platform.

Business impact: The client can now avoid the cost of AI outages and have confidence that AI functions will have the scalability and resiliency required for enterprise operations, in tandem with other critical infrastructure operations.

Example solution:

- Data/Application: Mongo DB
- AI Platform/Framework: TensorFlow

Insurance

Claims Image Analysis

Business scenario: A client wants to predict whether an image of minor car accident damage is a real claim, and whether it is similar to other claims that have been previously paid.

Business impact: The client can now achieve improved speed of assessments and reduced time spent for manual follow ups by prioritizing cases based on risk assessments.

Example solution:

- Data/Application: PostgreSQL
- AI Platform/Framework: PyTorch, ONNX, DLC via AI Toolkit for IBM Z

Operational efficiency and costs

Business scenario: A global health insurer wants to analyze large volumes of medical records to predict dis-ease risk while adhering to HIPAA regulatory compliance.

Business impact: The client can now quickly predict high risk scenarios, while reducing data center hardware footprints and related energy consumption. The overall impact is a reduction in cost for space and power.

Example solution:

- Data/Application: MongoDB
- AI Platform/Framework: PyTorch, ONNX, DLC via AI Toolkit for IBM Z

Government

Private data chatbot

Business scenario: A government client would like to offer automated services for federal benefits manage-ment. The desired result is chatbot-enabled automation of repetitive tasks, with personalized user experiences to help resolve questions and complete basic account updates on private data.

Business impact: The client can now achieve automation of manual, repetitive tasks resulting in reduced opera-tional costs and improved user satisfaction. All of these also meet regulatory standards.

Example solution:

- Data/Application: MongoDB
- AI Platform/Framework: TensorFlow

Reduce risk of error and operational costs

Business scenario: A federal client wants to consider family relationships at immigration and customs as part of predictive risk for criminal concerns. Even a reasonably accurate model with machine learning and natural lan-guage processing would decrease the number of manual calls and checking across names, misspellings and languages.

Business impact: The client can now save cost of space and power, as well as reduce cost of manual review.

Example solution:

- Data/Application: MongoDB
- AI Platform/Framework: TensorFlow

Wholesale Distribution

Variable pricing trends

Business scenario: A client wants to predict pricing optimization opportunities region-by-region, with incoming supply volumes, seasonal information, and types of goods.

Business impact: The client can now implement pricing contracts that include a variable range, or "market val-ue" for key product types. Market value can be determined with automated governance model input that is au-ditable to regulators.

Example solution:

- Data/Application: MongoDB

- AI Platform/Framework: PyTorch, ONNX, DLC via AI Toolkit for IBM Z

Improving partner onboard

Business scenario: A client wants to predict if a new vendor or product is a good fit to join the partner network by identifying similar vendors, or complementary products based on Know Your Customer information for buy-ers.

Business impact: The client can now run a batch workload that polls past transactional data and can highlight similar or dissimilar vendor patterns across different buyers/regions.

Example solution:

- Data/Application: PostgreSQL
- AI Platform/Framework: PyTorch, ONNX, DLC via AI Toolkit for IBM Z

Education

AI Training Classes

Business scenario: A client wants to offer classes to improve AI prediction models for fraud detection, anti money laundering, and other high transaction rate or highly sensitive data in an optimized and sustainable way.

Business impact: The client can now take advantage of scalability benefits, alongside Linux industry standard applications for running sample fraud detection and AML standards with synthetic data sets.

Example solution:

- Data/Application: PostgreSQL
- AI Platform/Framework: PyTorch, ONNX, DLC via AI Toolkit for IBM Z

Reduce Risk of Loss

Business scenario: A client wants to improve student satisfaction with payroll and student information systems. AI models are envisioned to identify when information does not appear correctly due to unexpected values.

Business impact: The client can now significantly reduce the number of errors and manual follow ups required. In addition, this also improves student satisfaction.

Example solution:

- Data/Application: DB2 LUW
- AI Platform/Framework: Cloud Pak for Data

4.5 Additional solution templates and resources to get started with AI on Linux

While this chapter focuses on two examples of multiple AI model solutions, there are other patterns that can be leveraged for implementation of other business use cases.

The AI on IBM Z team, in collaboration with the Open Mainframe Project at the Linux Foundation as part of the Ambitus project, is offering solution templates for various AI on Z use cases. These AI Solution Templates can be leveraged to accelerate the development, deployment, and production planning of AI models and applications on IBM Z and LinuxONE.

- For a template for Fraud Detection using the AI Toolkit for IBM Z and LinuxONE, see: https://github.com/ambitus/aionz-st-fraud-detection-tis?tab=readme-ov-file
- For other AI solution templates, see: https://ibm.github.io/ai-on-z-101/solutiontemplates/
- Contact the AI on IBM Z team (aionz@us.ibm.com) for other available options

Concepts and inspirations from AI solution templates that target z/OS are also relevant for Linux on IBM Z and LinuxONE.

REDP-5756-00

ISBN

Printed in U.S.A.