

Effortlessly Accelerate Your AI Applications and Workflows on Red Hat OpenShift Container Platform with 4th Gen Intel® Xeon® Scalable Processors

Intel's extensions of Red Hat OpenShift validated patterns take advantage of Intel® AMX to enhance AI training and inference workloads

Contents

Executive Summary	1
Solution Brief	2
Solution Architecture Highlights	2
Use Cases	3
Revolutionizing Retail with Multicloud on Red Hat OpenShift	4
Enhancing Medical Diagnosis with AI on Red Hat OpenShift	5
Summary	6
Learn More	6

Solution Benefits

- Easily deployable, fully functional, end-to-end workflows on the Red Hat® OpenShift® Container Platform can be created with no concern for configuration details.
- No-hassle enablement of the built-in AI accelerator on 4th Gen Intel® Xeon® Scalable processors.
- Fast inference and training for a variety of AI workloads at the edge.

Executive Summary

AI is being woven into nearly every field of endeavor. But sometimes, it can be overwhelming to know where to start to enable the full potential of AI. Red Hat and Intel are working together to simplify the deployment of hybrid cloud AI workloads.

Red Hat has developed validated patterns—all the code and Red Hat® OpenShift® Container Platform elements you need to deploy specific use cases—including recommender engines in retail and AI-powered image analytics for medical diagnosis. Validated patterns are continuously tested, and new ones are regularly added.

Intel has worked with Red Hat to show how easy it is to extend validated patterns to take advantage of specific hardware accelerators built into 4th Generation Intel® Xeon® Scalable processors. One such accelerator is Intel® Advanced Matrix Extensions (Intel® AMX), which is purpose-built to accelerate AI inference and training. With just a few extra lines of code and the use of Red Hat's Node Feature Discovery (NFD) operator, Intel extended the Multicloud GitOps validated pattern to take advantage of Intel AMX—potentially accelerating inference by as much as 10x compared to previous-generation Intel Xeon Scalable processors.¹

In another example of validated pattern extension, Intel quantized a medical diagnosis machine-learning model to use a lower precision to accelerate image inference at the edge. Quantization is a technique used in AI to reduce the computational and memory costs of running inference.² Using the NFD operator to identify nodes equipped with Intel AMX and an open-source scaling tool, the extended validated pattern unleashed the power of AI to assess chest X-rays for the risk of pneumonia.

These extended validated patterns are available on GitHub to use as-is or as a foundation for further extensions. Red Hat's robust and flexible OpenShift platform, combined with continued innovation from Intel and the developer ecosystem, demystifies deploying and accelerating AI workloads.



Solution Brief

Business Challenge

Many industries, including retail and healthcare, are increasingly using AI in their workloads. Experts predict that 70% of businesses will use AI by 2030.³ From personalized product recommendations to AI-powered disease diagnosis, the opportunities for AI to transform businesses and help save lives are enormous. Another industry trend is the move to cloud-native applications running on platforms like the latest Red Hat® OpenShift® Container Platform with Kubernetes. However, developers face several challenges when using AI and hybrid cloud deployments.

- Kubernetes environments are complex to set up from scratch, with many platform components and considerations.
- Code optimizations, like AI acceleration, are scattered across various sources and repositories, making them difficult to find and validate.

What if there was a way to quickly deploy a workload on the Red Hat OpenShift Container Platform, with minimal configuration and management overhead—plus easy access to optimizations that enable fast AI training and inference in the data center or at the edge?

Solution Value

The latest Red Hat OpenShift Container Platform provides DevOps teams and IT organizations with a hybrid cloud application platform for deploying new and existing edge applications on secure, scalable resources. But while the Red Hat OpenShift Container Platform is a powerful tool, it can be difficult to determine all the right components for a particular use case and workload.

Red Hat OpenShift validated patterns let you quickly create fully functional, end-to-end workflows on Red Hat OpenShift without worrying about configuration details and workflow management. Validated patterns are designed to connect multiple clouds and clusters, including edge clusters, can help you easily develop a solution that fulfills requirements and drives business success. Think of a validated pattern as a trusted recipe; you don't have to determine what ingredients are necessary or wonder if the recipe actually results in something edible—it just works.

Intel and Red Hat have collaborated to optimize Red Hat OpenShift and some of its validated patterns for 4th Generation Intel® Xeon® Scalable processors. Red Hat engineers maintain validated patterns to ensure they meet performance, scalability and reliability requirements and include all the necessary Red Hat OpenShift Container Platform components, as well as management software. They also automatically handle system configuration, security, networking, storage and application deployment. Validated patterns can be extended to enable the Red Hat OpenShift Container Platform to use additional hardware features by using Red Hat's Node Feature Discovery (NFD) operator. For example, Intel® Advanced Matrix Extensions (Intel® AMX) is a built-in AI accelerator on 4th Gen Intel

Xeon Scalable processors that can provide up to 10x faster training and inference compared to the previous generation of Intel Xeon Scalable processors.⁴ Image and video recognition, scientific computing, financial modeling, healthcare, natural language processing and many more use cases can benefit from using Intel AMX to quickly uncover insights, increase customer satisfaction and potentially lower compute costs.

This reference architecture illustrates how easy it is to enable Intel AMX and gain its performance advantages.



10x FASTER

training and inference

compared to the previous generation of Intel® Xeon® Scalable processors¹

Solution Architecture Highlights

Red Hat's validated patterns contain all the code needed to help build your technology stack so that you can bring solutions to life more quickly. All the steps are fully automated through GitOps processes to automate deployments consistently and at scale. Moreover, unlike static reference architectures, Red Hat validated patterns are continuously updated and refined based on customer feedback, industry trends and technological advancements. As an example of how the IT community can contribute to continuous innovation, with just a few lines of code, Intel has customized two Red Hat validated patterns to enable the use of Intel AMX (see Figure 1).

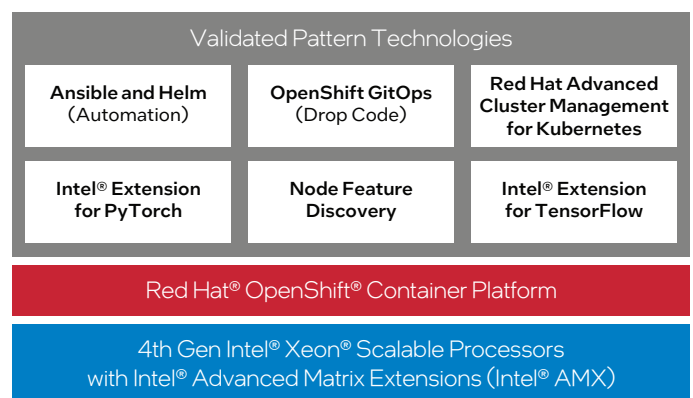


Figure 1. Extending Red Hat® validated patterns to include Intel® AMX can accelerate AI workloads.

A Closer Look at Intel's Extension of Red Hat Validated Patterns

The following sections briefly describe some of the most relevant technologies from Red Hat and Intel that power the validated pattern use cases illustrated in this reference architecture.

Red Hat OpenShift

Red Hat OpenShift offers a comprehensive platform for efficiently building, updating and scaling applications. You can enhance productivity and speed up your application deployment process by utilizing a full suite of services, all adaptable to your preferred infrastructure.

Red Hat OpenShift Operators

Red Hat OpenShift Container Platform relies heavily on operators, which are a group of software extensions and tools used to automate, manage and simplify the deployment and operation of complex applications and services. Software suppliers and developers often create operators with in-depth knowledge of a specific application or service.

The **NFD operator** controls the detection of hardware characteristics and configuration in a Red Hat OpenShift Container Platform cluster and labels the nodes with hardware-specific information. By marking nodes, it indicates the presence of a technology, accelerator or module on the host. By determining which nodes have a required hardware feature, DevOps teams can ensure that workflows requiring a specific feature (such as Intel AMX) land on an appropriate node.

Knative Serving

To declare and control how serverless applications behave within the cluster, **Knative Serving** creates a set of Kubernetes Custom Resource Definitions (CRDs). Its purpose is to automatically scale the number of containers up or down and assign them to nodes with required labels.

Argo CD

Argo CD is a declarative GitOps continuous delivery tool for Kubernetes that uses Git as the sole source of truth. It manages application definitions, configurations and environment versions in a declarative and automated manner while controlling component versions. Argo CD makes deploying and managing applications easier, faster and less problematic.

Argo CD is used in both of the extended validated patterns described in this reference architecture.

Intel AMX

Intel AMX is a built-in accelerator that enables 4th Gen Intel Xeon Scalable processors to optimize deep-learning (DL) training and inferencing workloads. With Intel AMX, 4th Gen Intel Xeon Scalable processors can quickly pivot between optimizing general computing and AI workloads. Imagine an automobile that could excel at city driving and quickly change to deliver Formula 1 racing performance. 4th Gen Intel Xeon Scalable processors deliver this type of flexibility. Developers can code AI functionality to take advantage of the Intel AMX instruction set, and they can code non-AI functionality to use the processor instruction set architecture. Intel has integrated the Intel® oneAPI Deep Neural Network Library (oneDNN) into popular open-source tools for AI applications, including TensorFlow, PyTorch, PaddlePaddle and ONNX.

Use Cases

In this document, we describe how Intel extended two validated patterns—Multicloud GitOps and Medical Diagnosis—to take advantage of Intel AMX performance gains.

Important note: The Linux kernel detects Intel AMX at run-time, so it is unnecessary to enable and configure it separately. For both patterns, the NFD operator is deployed to allow easy detection and consumption of Intel features and accelerators such as Intel AMX.

After pattern deployment, you can validate Intel AMX availability on the nodes. Log in to the Red Hat OpenShift Container Platform cluster and run the following command:

```
$ oc get nodes --show-labels
```

If the NFD operator was deployed successfully, nodes equipped with Intel AMX are labeled as shown below:

```
feature.node.kubernetes.io/cpu-cpuid.AMXBF16=true  
feature.node.kubernetes.io/cpu-cpuid.AMXINT8=true  
feature.node.kubernetes.io/cpu-cpuid.AMXTILE=true
```



Revolutionizing Retail with Multicloud on Red Hat OpenShift

Retailers can use AI and the Red Hat OpenShift Container Platform to significantly boost sales by deploying recommendation engines. These can run in the data center (think recommendations for more movies to watch after viewing Peter Pan or Spiderman) or they can run at the edge (imagine viewing product recommendations displayed on your supermarket shopping cart based on your purchase history and location in the store). This use case requires a platform that is agile enough to work on-premises or in the cloud, and that has modern integrated tools to make the transition to containers fast and easy. Red Hat and Intel are dedicated to making a flexible platform that lets teams gain the most benefit from AI.

[Multicloud GitOps](#) is one of the simplest Red Hat OpenShift Container Platform validated patterns, but provides a strong foundation to build upon whether the goal is a data center application or fast AI at the edge. This pattern was created to manage hybrid and multicloud deployments simply and securely. It is based on the GitOps (infrastructure-as-code) approach. The principle is to control and automate the deployment and management of applications and infrastructure using Git as a single source of truth. This section describes how the Multicloud GitOps validated pattern can accelerate time to value using AI with Intel AMX. The technique demonstrated here is easily transferable to any other validated pattern. It is also a straightforward way of using Intel AMX and proves that running advanced AI doesn't have to be complicated.

Multicloud GitOps with Intel AMX

In this use case, we deployed a managed application called `amx-app`. It works with the Deep Interest Evolution Network (DIEN), which is a machine-learning model used in the domain of personalized content recommendation. DIEN is designed to improve the accuracy of suggestions in scenarios where user interests evolve, such as in e-commerce platforms or content streaming services.

One of the advantages of using Intel AMX is its ability to run inference workloads at precisions lower than FP32. Lower precisions use smaller chunks of memory and can lead to accelerated processing,⁵ meaning that the retail customer experience is enhanced through faster delivery of product recommendations. In general, recommendation engines help increase overall yearly sales by about 20% and increase annual core corporate profitability by roughly 30%.⁶

The `amx-app` runs DIEN inference using the Intel[®] Optimization of TensorFlow and measures DIEN's accuracy for the `bfloat16` (BF16) precision. As mentioned earlier, we used the NFD operator to label nodes that support Intel AMX.

To deploy the Multicloud GitOps with Intel AMX validated pattern, follow the instructions on the [pattern's website](#). Next, validate Intel AMX availability using the commands provided earlier.

Implementation Details

The `amx-app` was deployed using instructions from the Model Zoo for Intel[®] Architecture repository and uses the `intel/recommendation:tf-spr-dien-inference` image on [Dockerhub](#). This image requires root access; obtaining that permission requires an update to the `anyuid` Security Context Constraints. To avoid time-consuming manual granting of permissions, we created YAML files describing `ClusterRole` and `RoleBinding`, among others.

- `ClusterRole` contains rules that allow using the root access.
- `RoleBinding` binds the new `ClusterRole` to the existing default service account.

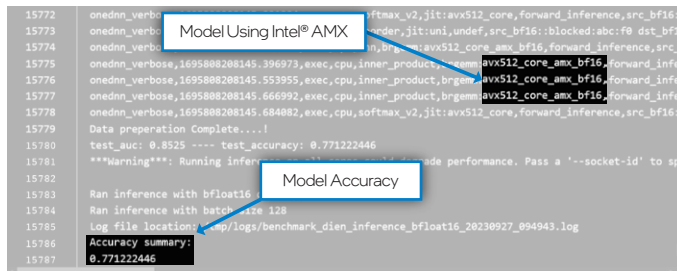
The `init-container` downloads the dataset and prepares it for use. We use a persistent volume claim to store data. Once the dataset is prepared, the pod starts and runs the `accuracy.sh` script to measure the inference accuracy. When the script completes, it prints the accuracy value compared to the inference results based on a labeled set of data:

```
containers:
  - name: run-inference
    env:
      - name: DATASET_DIR
        value: /tmp/dataset
      - name: OUTPUT_DIR
        value: /tmp/logs
      - name: SCRIPT
        value: accuracy.sh
      - name: PRECISION
        value: bfloat16
      - name: ONEDNN_VERBOSE_TIMESTAMP
        value: '1'
      - name: ONEDNN_VERBOSE
        value: '1'
    volumeMounts:
      - name: dataset
        readOnly: true
        mountPath: /tmp/dataset
    image: 'intel/recommendation:tf-spr-dien-inference'
    args:
      - /bin/bash
      - '-c'
      - /bin/bash quickstart/${SCRIPT}; sleep 24h
```


To ensure that the pod containing the `amx-app` is scheduled on a node equipped with Intel AMX, in the deployment we added the following selector:

```
spec:
  nodeSelector:
    feature.node.kubernetes.io/cpu-cpuid.AMXBF16:
      'true'
```

By enabling `ONEDNN_VERBOSE` in the `accuracy.sh` script, all the compiled instructions are shown in the logs. The appearance of `"avx_512_core_amx_bf16"` confirms that Intel AMX is used.



```
15772 onednn_verbose,1695888208145.396973,exec,cpu,inner_product,brgemv,avx512_core_amx_bf16,forward_infi
15773 onednn_verbose,1695888208145.400000,loader,jit:uni,undef,src_bf16::blocked:abc:00_dst_bf16
15774 onednn_verbose,1695888208145.400000,inner_product,brgemv,avx512_core_amx_bf16,forward_inference,src
15775 onednn_verbose,1695888208145.400000,inner_product,brgemv,avx512_core_amx_bf16,forward_infi
15776 onednn_verbose,1695888208145.553955,exec,cpu,inner_product,brgemv,avx512_core_amx_bf16,forward_infi
15777 onednn_verbose,1695888208145.666992,exec,cpu,inner_product,brgemv,avx512_core_amx_bf16,forward_infi
15778 onednn_verbose,1695888208145.684802,exec,cpu,softmax_v2,jit:avx512_core,forward_inference,src_bf16:
15779 Data preparation Complete...!
15780 test_auc: 0.8525 ---- test_accuracy: 0.77122446
15781 ***Warning***: Running inference on a node without Intel AMX hardware. Pass a '--socket-id' to s
15782
15783 Ran inference with bfloat16
15784 Ran inference with batch size 128
15785 Log file location: /tmp/logs/benchmark_dien_inference_bfloat16_20230927_094943.log
15786 Accuracy summary:
15787 0.77122446
```

Figure 2. Log data from the `amx-app` pod.



Enhancing Medical Diagnosis with AI on Red Hat OpenShift

The [Medical Diagnosis](#) validated pattern runs on a Red Hat OpenShift Container Platform cluster with an AI-powered image classification workload. This pattern, used at the edge, is an automated data pipeline. It ingests real-time image data from multiple IoT sources, runs inference on the images in the medical clinic and securely stores the images. The anonymized data is sent to the data center (in the lab) for labeling and model retraining.

This solution helps healthcare providers set up their clinical environment to take in data from a myriad of medical IoT devices in the modern medical office. For example, the pattern can be used to run an application that performs chest X-ray analysis. A preconfigured machine-learning model makes a risk assessment for pneumonia shown in images. All results, alongside metrics collected from Prometheus, are displayed in a Grafana dashboard in real time. As mentioned earlier, we used the NFD operator to label nodes that support Intel AMX.

To deploy the Medical Diagnosis with the Intel AMX validated pattern, follow the instructions on the [pattern's website](#). Next, validate Intel AMX availability using the commands provided earlier.

Implementation Details

To showcase the full accelerated inference capabilities of Intel AMX, we customized the machine-learning model used in the Medical Diagnosis validated pattern—we used the [Intel® Neural Compressor](#) to quantize the original Keras machine-learning model to INT8 precision. The resulting quantized model is delivered in TensorFlow format. The first step in quantizing the model was to convert the Keras model to the TensorFlow SavedModel format:

```
from tensorflow.keras.models import load_model
model = load_model("pneumonia_model.h5")
model.save("pneumonia_model")
```

Then, we prepared a calibration dataset made of X-ray images, which were primarily used to train the machine-learning model:

```
from tensorflow.keras.preprocessing.image import
ImageDataGenerator
img_width, img_height = 150, 150
batch_size = 16

test_datagen = ImageDataGenerator(rescale=1. / 255)

test_generator = test_datagen.flow_from_directory(
    validation_data_dir,
    target_size=(img_width, img_height),
    batch_size=batch_size,
    class_mode='binary')
```

The last step of the quantization was to use the `fit` command from the Neural Compressor and save the new model:

```
from neural_compressor.quantization import fit
neuralcomp_model = fit(model='pneumonia_model'),
conf=config, calib_data_loader=test_generator)

neuralcomp_model.save('pneumonia_model_quantized')
```

Knative Serving scales containers with the machine-learning model. Knative Serving was configured to schedule pods on nodes where Intel AMX was available.

Once setup is complete, the user initiates the pattern workflow by starting the image generator component; set the number of replicas to 1 using the following command:

```
oc scale deploymentconfig/image-generator
--replicas=1 -n xraylab-1
```

After the image flow has started, Knative Serving automatically creates pods with the model to process incoming images. The number of generated images can be changed by scaling the number of image generator pods up or down. The whole workflow, alongside images processed in real time, can be seen in the Grafana dashboard.

Summary

Technology is a wonderful thing, and it changes quickly. Keeping up can be challenging. Intel and Red Hat are working together to inspire DevOps teams to access the latest innovations. Red Hat validated patterns make it easy to bring those innovations to bear on real-world use cases. Intel is excited to work with latest Red Hat on OpenShift Container Platform to bring a robust enterprise platform that can simplify today's technology challenges. Starting with these two validated pattern options (Multicloud GitOps and Medical Diagnosis), we encourage you to take advantage of accelerated compute with Intel AMX, whether you work in the manufacturing, healthcare or other diverse fields like logistics, banking and finance.

Contact your Intel or Red Hat representative today to discuss how Intel AMX can benefit your business.

Learn More

- [4th Gen Intel® Xeon® Scalable processors](#)
- [Intel® Advanced Matrix Extensions \(Intel® AMX\)](#)
- [Red Hat validated patterns](#)

Revision History

Revision Number	Description	Date
1.0	First Release	January 2024



Red Hat

intel

¹ See [A16] and [A17] in the [Performance Index](#). 4th Gen Intel Xeon Scalable processors. Results may vary.

² Quantization refers to techniques for performing computations and storing tensors at lower bitwidths than floating-point precision. A quantized model executes some or all of the operations on tensors with reduced precision rather than full precision (floating-point) values. This allows for a more compact model representation and the use of high-performance vectorized operations on many hardware platforms.

³ The Motley Fool, October 2023, "70% of Companies Will Use AI by 2030 -- 2 Stocks You'll Want to Buy Hand Over Fist."

⁴ See endnote 1.

⁵ Intel, <https://www.intel.com/content/www/us/en/developer/articles/technical/accelerate-pytorch-training-inference-on-amx.html#gs.lsiw8k>.

⁶ McKinsey & Company, "Targeted online marketing programs boost customer conversion rates."

Performance varies by use, configuration and other factors. Learn more at www.Intel.com/PerformanceIndex.

Performance results are based on testing as of dates shown in configurations and may not reflect all publicly available updates. See backup for configuration details. No product or component can be absolutely secure.

Your costs and results may vary.

Intel technologies may require enabled hardware, software or service activation.

© Intel Corporation. Intel, the Intel logo, and other Intel marks are trademarks of Intel Corporation or its subsidiaries.

Other names and brands may be claimed as the property of others. 0124/JCAP/KC/PDF