Quick Start Guide

Network and Edge Reference System Architectures - On Premises Edge AI Box

Develop and verify edge analytics services for On Prem Edge AI Box using BMRA on the Intel® Core™ processor.

Authors

Abhijit Sinha

Zhifang Long

Alex Lam

Introduction

The Reference System Architectures (Reference System[1]) are forward-looking template solutions for fast automated software installation and deployment.

This document is a quick start guide to configure and deploy **Edge AI Box** underlying software requirements using the **Container Bare Metal Reference System Architecture (BMRA)** on **Intel® Core™ processors** with either **Intel® Arc™ Discrete Graphics GPU** or **Intel® Iris® Xᵉ Integrated Graphics** platform.

The Reference System is deployed using the **On Prem Edge AI Box Configuration Profile** with **optimized configuration for edge video analytics workloads in a single box** in real time for lightweight edge devices. Video Analytics is enabled by OpenVINO™ toolkit and a choice of OpenCV or Intel® Deep Learning Streamer (Intel® DL Streamer) as AI-based media analytics frameworks. The platform is accelerated by Intel® Arc™ Discrete Graphics GPU or Intel® Iris® Xᵉ Integrated Graphics, as shown in Figure 1.

On Prem Edge AI Box Architecture

Figure 1 shows the architecture diagram of the On Prem Edge AI Box Profile where media analytics frameworks OpenCV and Intel® DL Streamer are containerized and work alongside a Video Analytics base library container including OpenVINO™ toolkit and media accelerators, and drivers. The provided container suite is used for microservice-based system architectures.

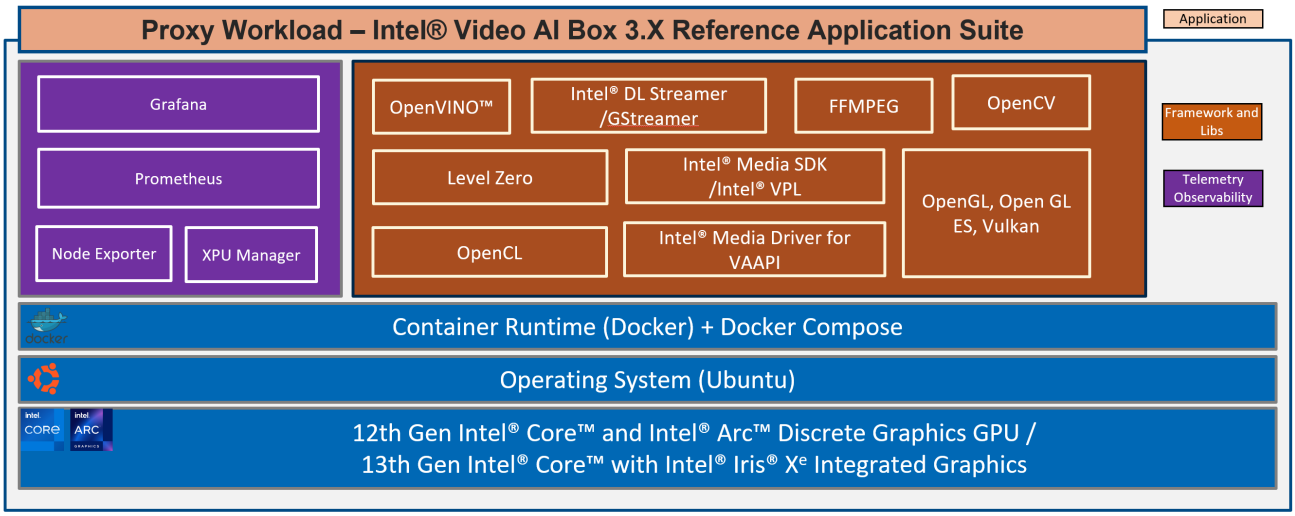


Figure 1: Architecture of Edge AI Box deployment using BMRA `on_prem_aibox` Profile

[1] In this document, "Reference System" refers to the Network and Edge Reference System Architecture.

Hardware BOM

Following is the list of the hardware components that are required for setting up Reference Systems:

Ansible Host	Laptop or server running a UNIX base distribution
Target Node	1x 11th Gen Intel® Core™ mobile processor with Intel® Iris® Xᵉ Integrated Graphics; OR 1x 12th Gen Intel® Core™ desktop processor with Intel® Arc™ Discrete Graphics GPU; OR 1x 12th Gen Intel® Core™ mobile processor with Intel® Iris® Xᵉ Integrated Graphics; OR 1x 12th Gen Intel® Core™ processor for IOT Edge with Intel® Iris® Xᵉ Integrated Graphics; OR 1x 13th Gen Intel® Core™ mobile processor with Intel® Iris® Xᵉ Integrated Graphics
GPU	Intel® Arc™ Discrete Graphics GPU A380 (only on 12th Gen Intel Core)
BIOS	Use the default BIOS settings (The user may need to disable secure boot to install the out of tree (OOT) drivers)

Software BOM

Following is the list of the software components that are required for setting up Reference Systems:

High Level Media Frameworks	Intel® DL Streamer, GStreamer, OpenCV, FFmpeg
Inference Frameworks	OpenVINO™ toolkit
Media and Video Acceleration	Intel® Media SDK/Intel® Video Processing Library (Intel® VPL), Intel® Media Driver for VAAPI, Libva
Graphics Compute Runtime	OpenGL, OpenCL, Level Zero GPU, GPU drivers
Observability	XPU Manager, Node Exporter, Prometheus, Grafana
Container Runtime	Docker, Docker-compose
OS	Ubuntu 22.04.2 Desktop (Kernel: 5.19)

For more details on software versions for the **On Prem Edge AI Box Profile**, refer to Chapter 4 of BMRA User Guide listed in the Reference Documentation section.

Getting Started

Prerequisites

Before starting the deployment, perform the following steps:

- A fresh OS installation is expected on the controller and target nodes to avoid a conflict between the RA deployment process with the existing software packages. To deploy RA on the existing OS, ensure that there is no prior Docker or Kubernetes* (K8s) installations on the server(s).
- The target nodes hostname must be in lowercase, numerals, and hyphen ' - '.
    - For example: wrk-8 is acceptable; wrk_8, WRK8, Wrk^8 are not accepted as hostnames.
- The target node must be Network Time Protocol (NTP) synced, i.e., correct date and time must be set.
- The BIOS on the target node is set as per the recommended settings.

Deployment Setup

Ansible playbooks are used to install the Bare Metal (BMRA), which sets up the infrastructure for an On Prem Edge AI Box. Figure 2 shows the deployment model for Edge AI Box infrastructure using BMRA.

The target device starts with Ubuntu 22.04.2 Desktop only, acting as both Ansible host and target, and it ends with the deployed infrastructure using the `on_prem_aibox` Reference System profile.

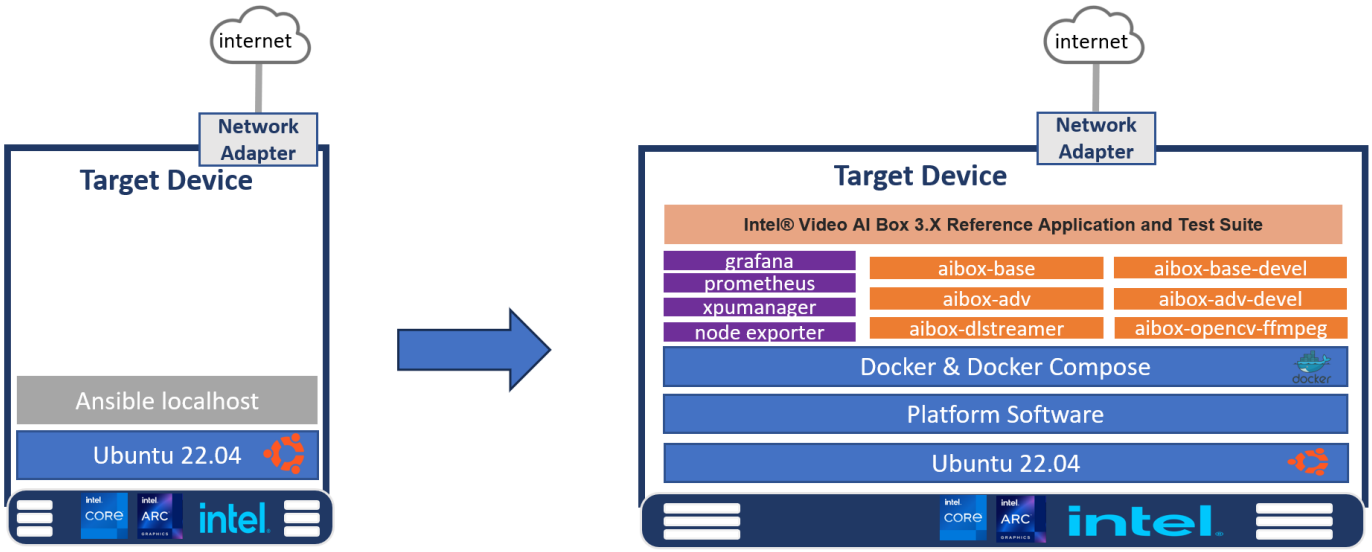


Figure 2: BMRA deployment setup for Edge AI Box

Installation Flow for RA Deployment

Ansible playbooks are used to install the Bare Metal (BMRA), which sets up the infrastructure for an On Prem Edge AI Box.

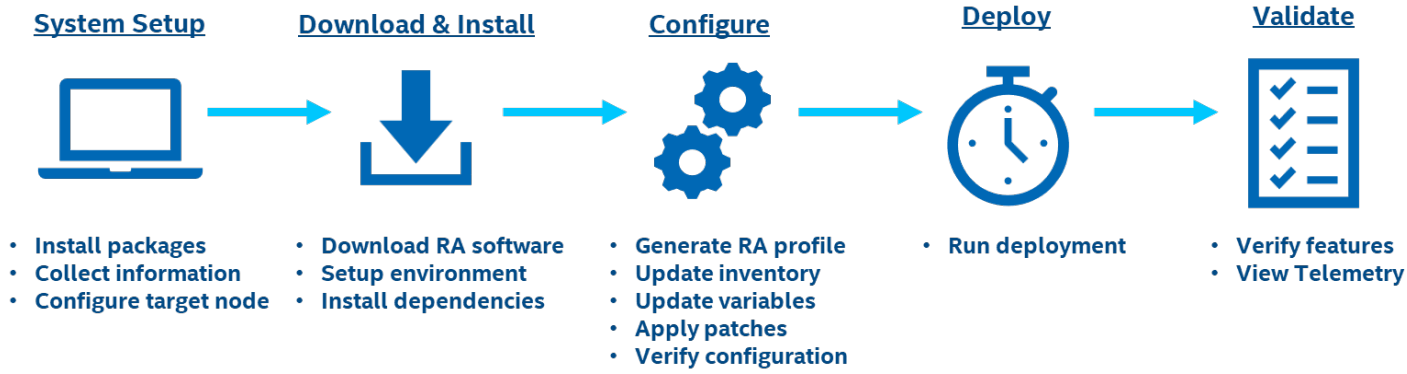


Figure 3: RA Deployment Flow

Step 1 - Set Up the System

The Edge AI Box is deployed on a single target host running Ubuntu OS. The deployment is on a localhost bare-metal environment (known as target host) and there is no need for a separate Ansible host for this deployment.

Target Host

System Setup

Install necessary packages (some might already be installed):

```
# sudo apt update
# sudo apt install -y python3 python3-pip openssh-client git build-essential
# pip3 install --upgrade pip
```

Step 2 - Download and Install

Target Host

Download & Install

1. Download the source code from the GitHub repository for the Reference System server:

```
# git clone https://github.com/intel/container-experience-kits/
# cd container-experience-kits
# git checkout v23.10
# git submodule update --init
```

2. Set up Python* virtual environment and install dependencies:

```
# python3 -m venv venv
# source venv/bin/activate
# pip3 install -r requirements.txt
```

3. Install Ansible dependencies for the Reference System:

```
# ansible-galaxy install -r collections/requirements.yml
```

Step 3 – Configure

Configure

The **On Prem AI Box** configuration profile (`on_prem_aibox`) is used for this deployment.

Target Host

1. Generate the configuration files:

```
# export PROFILE=on_prem_aibox
# make examples ARCH=core
# cp examples/k8s/${PROFILE}/inventory.ini .
```

Note: The AI Box is deployed on the target (localhost) so the *inventory.ini* file does not need updates.

2. Copy `group_vars` and `host_vars` directories to the project root directory:

```
# cp -r examples/k8s/${PROFILE}/group_vars examples/k8s/${PROFILE}/host_vars .
```

3. Update the `host_vars` filename with the target machine's hostname:

```
# mv host_vars/node1.yml host_vars/localhost.yml
```

4. If the server is behind a proxy, update *group_vars/all.yml* by updating and uncommenting the lines for `http_proxy`, https_proxy, and `additional_no_proxy`.

```
## Proxy configuration ##
http_proxy: "http://proxy.example.com:port"
https_proxy: "http://proxy.example.com:port"
additional_no_proxy: ".example.com,mirror_ip"
```

5. Apply required patches for Kubespray (Even though we do not install Kubernetes, it is needed for compatibility with other Ansible scripts):

```
# ansible-playbook -i inventory.ini playbooks/k8s/patch_kubespray.yml
```

6. (Recommended) You can check the dependencies of components enabled in `group_vars` and `host_vars` with the package dependency checker:

```
# ansible-playbook -i inventory.ini playbooks/preflight.yml
```

7. (Optional) Verify that Ansible can connect to the target server by running the following command and checking the output generated in the *all_system_facts.txt* file:

```
# ansible -i inventory.ini -m setup all > all_system_facts.txt
```

Step 4 – Deploy

Target Host

Deploy

Now the BMRA `on_prem_aibox` configuration profile can be deployed on the bare metal system by using the following command:

```
# ansible-playbook -i inventory.ini -b -K playbooks/on_prem_aibox.yml
```

Step 5 – Validate Video Analytics

Target Host

Validate

1. After the successful deployment of the `on_prem_aibox` profile, the base container-related Docker files and scripts are generated in the following location.

```
# ls /opt/intel/base_container/
      dockerfile/          # Base container Dockerfiles and build scripts
      test/                # Test container Dockerfiles, build scripts, and test
scripts
```

2. You can use the build and test scripts to build and test the base containers. Following is an example to build and test the `dlstreamer` base container. The test uses Intel® DL Streamer to detect cars in an input video.

```
# cd /opt/intel/base_container/dockerfile
# ./build_base.sh
# ./build_dlstreamer.sh

# cd /opt/intel/base_container/test
#./test_dlstreamer.sh
```

REPOSITORY	TAG	IMAGE ID	CREATED	SIZE
test-openvino	3.1	a4ae2b027ffd	19 hours ago	1.44GB
test-openvino	latest	a4ae2b027ffd	19 hours ago	1.44GB
test-openvino-adv-dev	3.1	38666d6f61d7	19 hours ago	12GB
test-openvino-adv-dev	latest	38666d6f61d7	19 hours ago	12GB
test-openvino-adv	3.1	131129162313	19 hours ago	1.45GB
test-openvino-adv	latest	131129162313	19 hours ago	1.45GB
test-opencv	3.1	8cb35cfaf625	19 hours ago	4.37GB
test-opencv	latest	8cb35cfaf625	19 hours ago	4.37GB
test-ffmpeg	3.1	85244ce42399	19 hours ago	4.37GB
test-ffmpeg	latest	85244ce42399	19 hours ago	4.37GB
test-dlstreamer	3.1	aa28820f4ae8	19 hours ago	3.29GB
test-dlstreamer	latest	aa28820f4ae8	19 hours ago	3.29GB
test-openvino-dev	3.1	632d74908e97	19 hours ago	9.69GB
test-openvino-dev	latest	632d74908e97	19 hours ago	9.69GB
aibox-adv-devel	3.1	35e6cee6b4b4	21 hours ago	12GB
aibox-adv-devel	latest	35e6cee6b4b4	21 hours ago	12GB
aibox-base-devel	3.1	34061a0c1cef	21 hours ago	9.69GB
aibox-base-devel	latest	34061a0c1cef	21 hours ago	9.69GB
aibox-adv	3.1	c45d4ab7a3c7	46 hours ago	1.45GB
aibox-adv	latest	c45d4ab7a3c7	46 hours ago	1.45GB
aibox-opencv-ffmpeg	3.1	0bf290da5502	46 hours ago	4.37GB
aibox-opencv-ffmpeg	latest	0bf290da5502	46 hours ago	4.37GB
aibox-dlstreamer	3.1	5133151d0285	2 days ago	3.29GB
aibox-dlstreamer	latest	5133151d0285	2 days ago	3.29GB
aibox-base	3.1	f7e257808c39	2 days ago	1.44GB
aibox-base	latest	f7e257808c39	2 days ago	1.44GB
grafana/grafana	10.1.2	31656ec60d2e	4 weeks ago	391MB
prom/prometheus	v2.47.0	9c703d373f61	6 weeks ago	245MB
prom/node-exporter	v1.6.1	458e026e6aa6	3 months ago	22.8MB
intel/xpumanager	v1.2.13	000cd3f12bf7	3 months ago	629MB
ubuntu	22.04	5a81c4b8502e	3 months ago	77.8MB

3. On test completion, the results can be checked. If the test is successful, you see `PASSED` in the test result file.

```
# cd ~/nep_validator_data/
# cat test_dlstreamer_result_aibox-dlstreamer
```

4. On successful test completion, the output video can be seen marked with rectangle bounding boxes and object labels in the `videos` directory:

```
# ls ~/nep_validator_data/videos
output_person-vehicle-bike-detection-2004.mp4
```

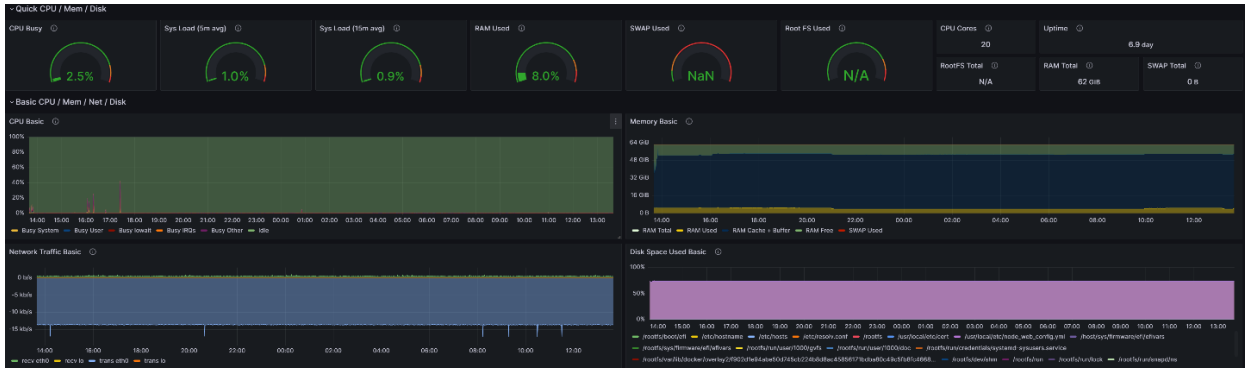
Figure 3: Edge AI Box test results with rectangle bounding boxes and object labels over the videos

Additional feature verification tests for the `on_prem_aibox` configuration profile can be found here.

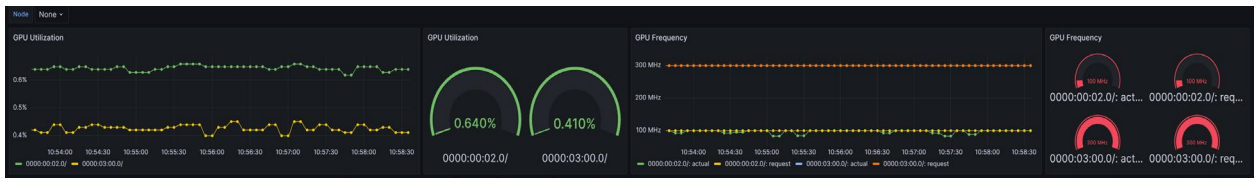
Step 6 – Validate Telemetry

1. After the successful deployment of the `on_prem_aibox` profile, telemetry related services are started automatically. User can use browser to open built-in dashboards to view the telemetry graphs.
   - If user uses the device with local display, they can directly login to https://127.0.0.1:3000/ with local browser.
   - If user uses the device remotely via ssh, they can forward port 3000 to a local PC by using the below command, then login to https://127.0.0.1:3000 with local browser.
     - "ssh -L 3000:127.0.0.1:3000 user@aibox_device_ip"

   The default username and password are admin/admin. The first login will ask user to change the password.
2. After login, click left top "menu" button, navigate to "Home -> Dashboards -> General", then you will see available dashboards.
   - Double-click node-exporter dashboard to see CPU and OS telemetries.



   - Double-click xpumanager dashboard to see GPU telemetries.



Reference Documentation

The *Network and Edge Bare Metal Reference System Architecture User Guide* provides information and a full set of installation instructions for a BMRA.

The *Network and Edge Reference System Architectures Portfolio User Manual* provides additional information for the Reference Architectures including a complete list of reference documents.

The *Edge AI Box* website provides more information for the sample test cases and usage for version 3.1.

Other collaterals, including technical guides and solution briefs that explain in detail the technologies enabled in the Reference Architectures are available in the following location: Network & Edge Platform Experience Kits.

Document Revision History

REVISION	DATE	DESCRIPTION
001	September 2023	Initial release.
002	October 2023	Updated BMRA version to 23.10 and added telemetry services to Edge AI Box.

1023/DN/WIT/PDF 788715-002US