

Dashboard for z/VM Neural Network Processor

z/VM Performance Data Pump

Behind the Dashboards

May 2025

Rob van der Heij
robvdheij@nl.ibm.com

z/VM Development – Endicott, NY

Neural Network Processing Assist

Specialized Function Assist introduced with z16

- Performs neural network tensor operations
- Repetitive operations on large matrices
- For z/VM exploited by Linux on IBM Z workload
- Operates on data in user (virtual machine) memory
- Integrates in memory cache architecture

Known Limitations

- This is a sample dashboard; clients must verify whether it is effective for their configuration and workload

This dashboard is licensed by IBM under the Apache 2.0 License and is provided 'as is' without warranty, representation, support, maintenance or an obligation to issue updates.

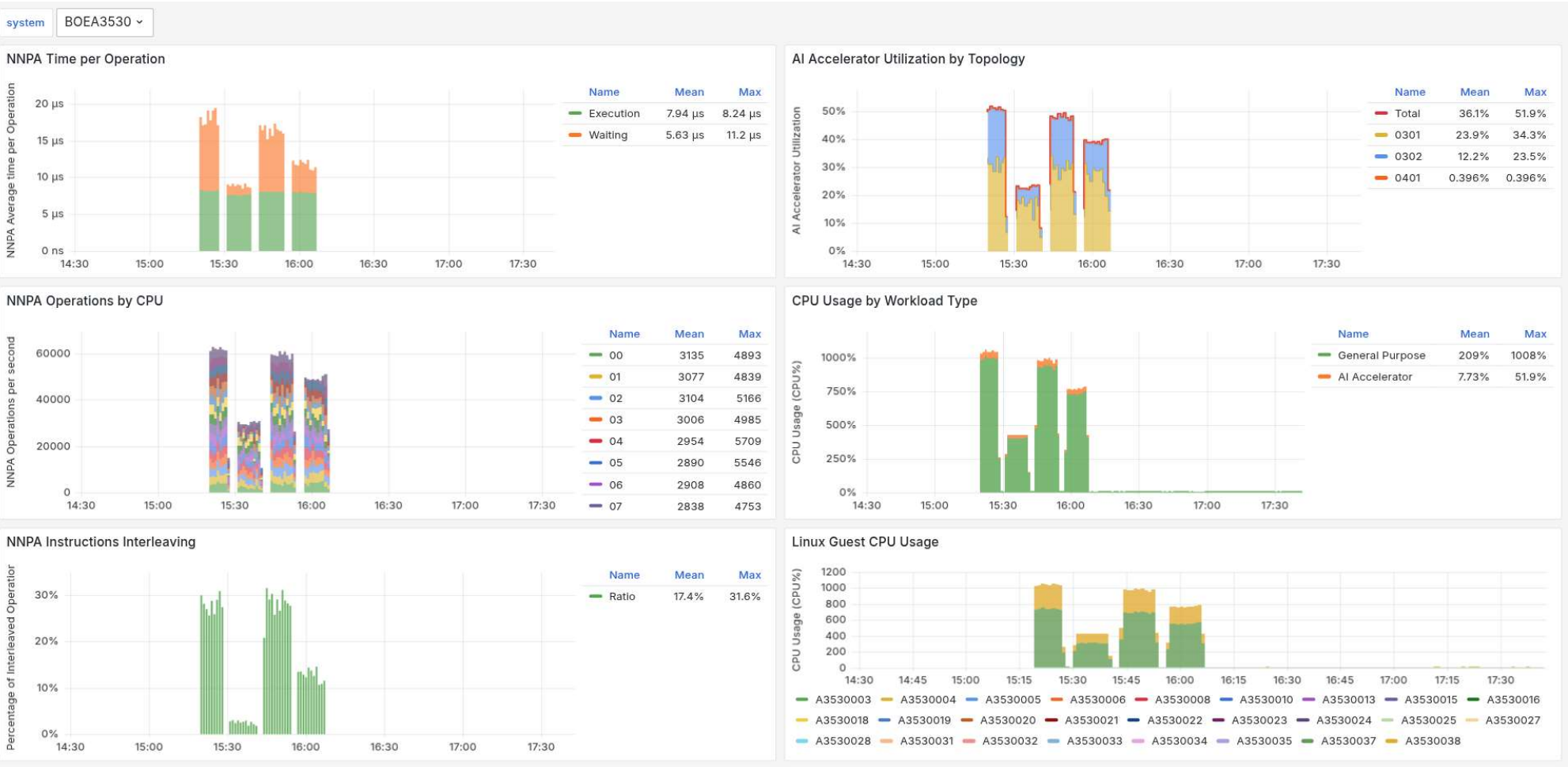
Invoked by NNPA general instruction

- Synchronous: processor is busy while AI Accelerator used
- Invokes AI Accelerator on same chip as the CPU core
- With z16 one AI Accelerator per chip - per 8 cores
- Different cores may be competing for the same AIU
- Dashboard shows utilization by AI Accelerator
- With z17 the AI Accelerator on other chips can be used

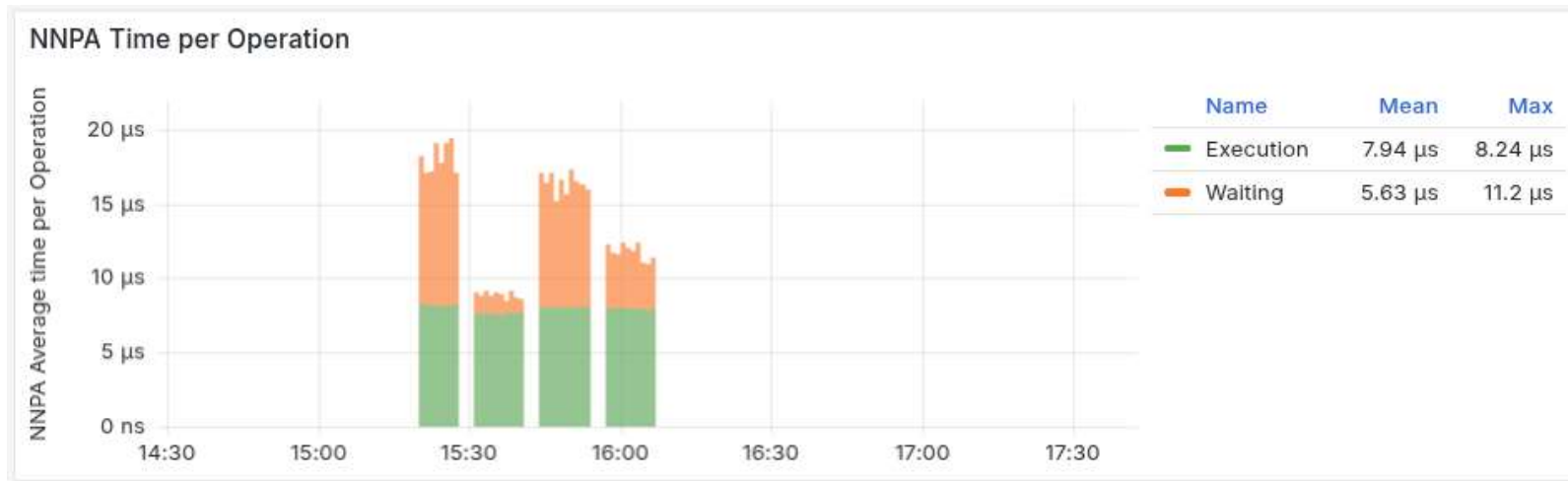
Instrumentation

- Metrics provided by firmware in CPUMF data
- Visible effects on other metrics in z/VM
- Published analytics counters do not apply to z/VM

Dashboard Overview



NNPA Time per Operation - z16



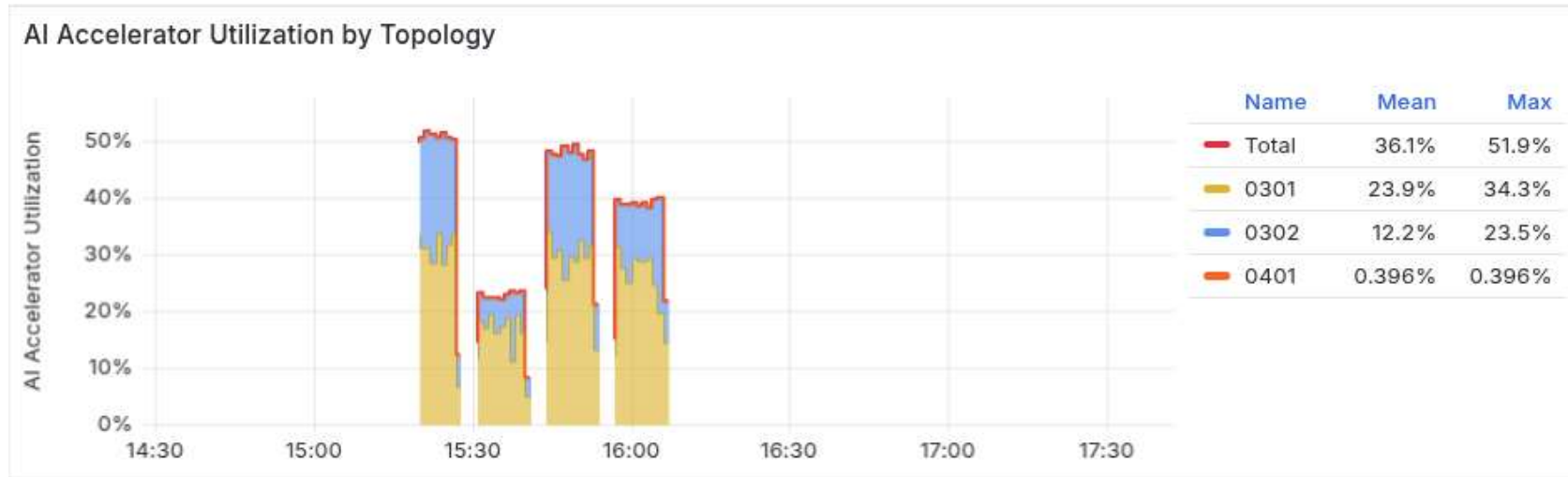
- The chart shows the average time used for an NNPA operation to complete. The time is measured by the CPU that issues the NNPA instruction that is held up while the AI Accelerator is performing the function.
- The "waiting" portion is when the CPU is held up while the AI Accelerator is executing an NNPA instruction for another CPU.
- We expect the "waiting" portion to increase when more CPUs are competing for the same AI Accelerator and utilization is high.
- The workload determines the mix of NNPA functions being used, the size of the model, and other aspects that affect the time it takes for the AI Accelerator to complete the required function.
- For long-running operations, the AI Accelerator will interrupt running operations and interleave execution of multiple NNPA instructions.

NNPA Time per Operation - z17



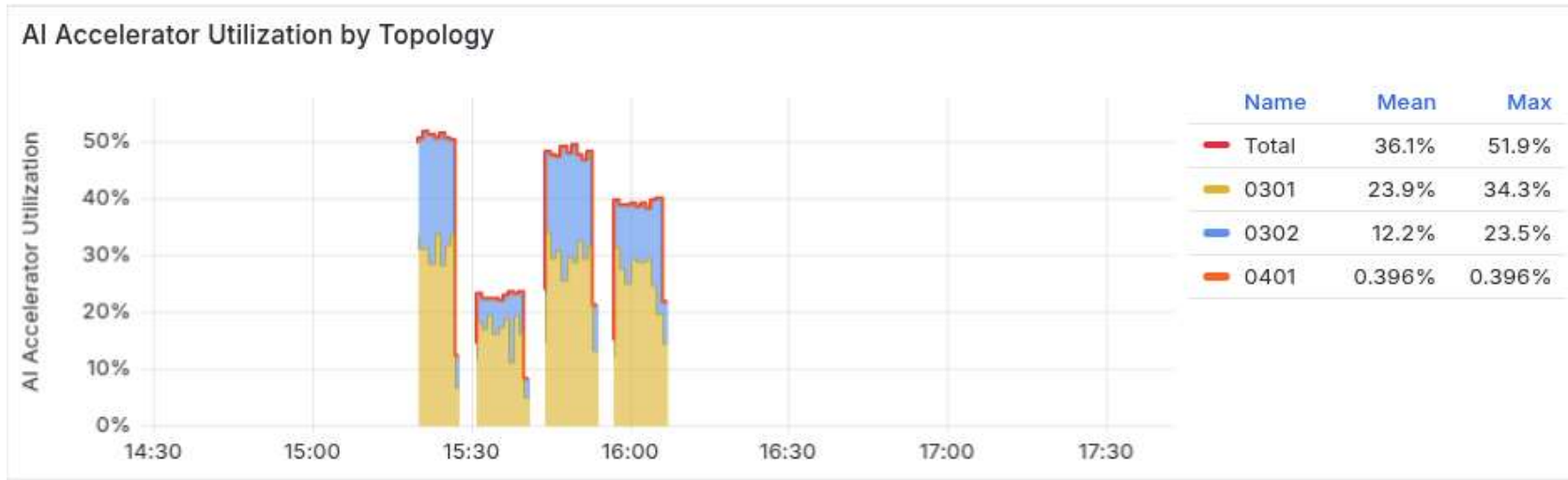
- With the Telum II processor in z17, an NNPA instruction can be performed by another chip when the local AI Accelerator is busy. Executing the NNPA instruction on a remote AI Accelerator greatly reduces the "waiting" time for access to the AI Accelerator. Depending on the CPC configuration, it is possible to see no waiting time anymore.
- When the NNPA instruction runs on another chip, data will be copied from the local cache to the remote cache. This leads to slightly slower processing of the AI Accelerator. The "Remote %" line in the chart shows the percentage of NNPA operations performed on another chip. It is expected that the "execution" time per NNPA operation will increase when more operations run on another chip.
- Users may want to measure their application with a low "Remote %" to have a baseline for NNPA execution time for their workload. The third test in the chart for example shows no remote operations and an average of 3.7 us, where 20% remote appears to increase it to 4.2 us. As often is the case with cache behavior, and illustrated by the chart, it depends.
- Selection of the AI Accelerator to use is done automatically by firmware and is not controlled much by user configuration. While some configuration choices may affect the number of local AI Accelerators available on z/VM, this is most likely not a leading factor of overall application performance.

AI Accelerator Utilization by Topology - z16



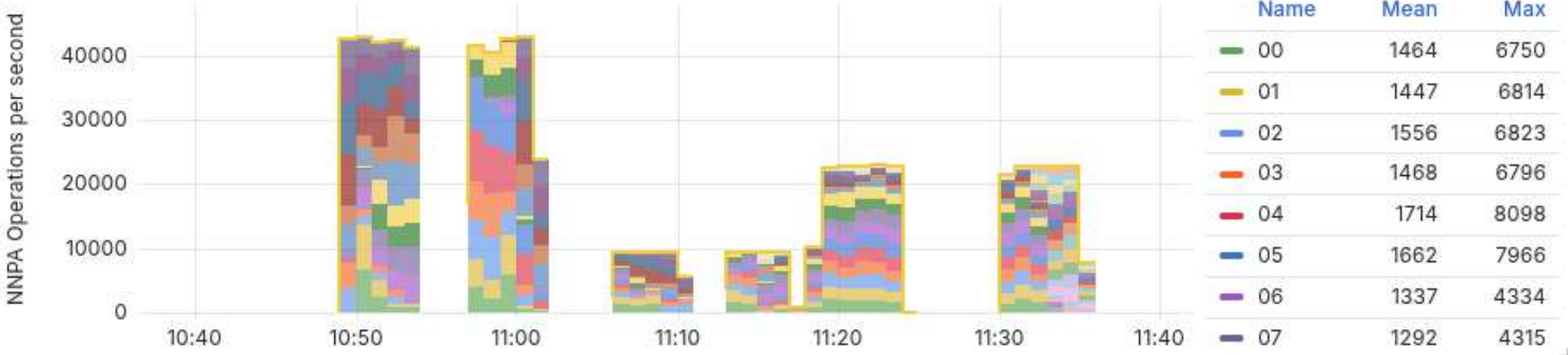
- The chart shows the utilization of each of the local AI Accelerators as well as the total combined usage of the local AI Accelerators. 100% would be one AI Accelerator fully utilized. When utilization of an AI Accelerator is high, we should expect contention and NNPA wait time.
- The total can show well above 100% when multiple AI Accelerators are engaged in the workload. More AI Accelerators are engaged in the workload when the work is spread over multiple CPUs, but other performance aspects benefit from a "vertical" configuration.
- The number of AI Accelerators engaged in the workload is determined by the LPAR configuration, the LPAR weight, the z/VM configuration and the Linux virtual machine configuration. Both PR/SM and z/VM try to keep virtual CPUs close together, so a single AI Accelerator on z16 is likely used by 8 Linux virtual CPUs (or 16, in case SMT-2 is used).
- Even when an AI Accelerator is used less than 100% by this z/VM LPAR, other LPARs with logical CPUs on the same chip may also be using the same AI Accelerator and thus show more wait time for this workload.

AI Accelerator Utilization by Topology - z17



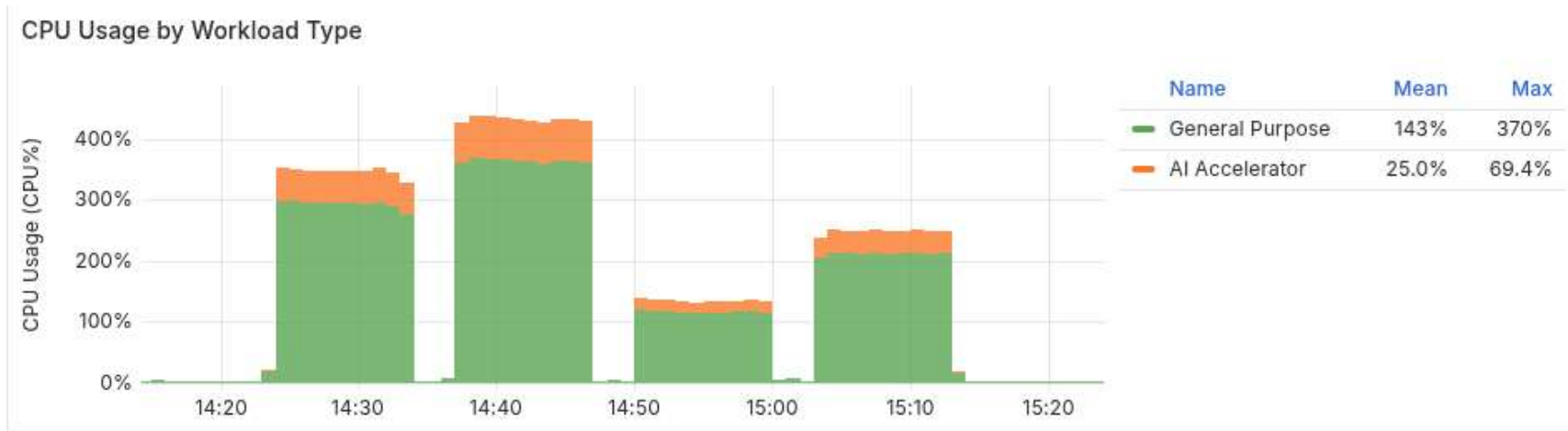
- On z17 an NNPA operation can use the AI Accelerator on another Telum II processor. Because the chart has grouped usage by local AI Accelerator on the same processor where the NNPA instruction is issued, the utilization of a single AI Accelerator in this chart can be more than 100% when remote NNPA operations are included.
- Since the workload in other LPARs may also run NNPA instructions on the local AI Accelerator of this z/VM LPAR, the actual utilization of the AI Accelerator may be higher than measured by instructions in this z/VM LPAR.
- It is reasonable to expect that when this LPAR finds that on average a significant percentage of NNPA operations is performed on remote AI Accelerators, that other AI workloads will on average also be using remote AI Accelerators. Since NNPA operations are only part of the application workload, the NNPA operations from other LPARs are expected to interleave well with this workload.

NNPA Operations by CPU



- The chart shows the average number of NNPA operations per second as issued by a logical CPU. The total shows the aggregated number of NNPA operations on all AI Accelerators.
- There is no detail about which NNPA functions are used, but for a given workload we would expect a consistent function mix so that the numbers give a good impression of the AI activity of the application.
- The chart shows both the total and the average per z/VM logical CPU. When more logical CPUs are involved, it is likely that more AI Accelerators in the configuration are being utilized. The utilization by AI Accelerator shows that breakdown.

CPU Usage by Workload Type



- Even applications that involve AI Accelerator functions will also perform a lot of non-AI operations to manipulate the data. The chart shows the portion of AI Accelerator usage in relation to the non-AI portion of the workload.
- Note that the "General Purpose" portion includes also the usage of other (Linux) virtual machines on the z/VM LPAR. When other unrelated workloads are running on the same z/VM system, the "Linux Guest CPU Usage" chart may be helpful to isolate the relevant workload.
- The chart shows both the total and the average per z/VM logical CPU. When more logical CPUs are involved, it is likely that more AI Accelerators in the configuration are being utilized. The utilization by AI Accelerator shows that breakdown.
- The benchmark workload in this chart shows around 15% of the total CPU seconds consist of AI Accelerator function. When unrelated workload runs on z/VM the "Linux Guest CPU Usage" chart may be useful to see the application workload.

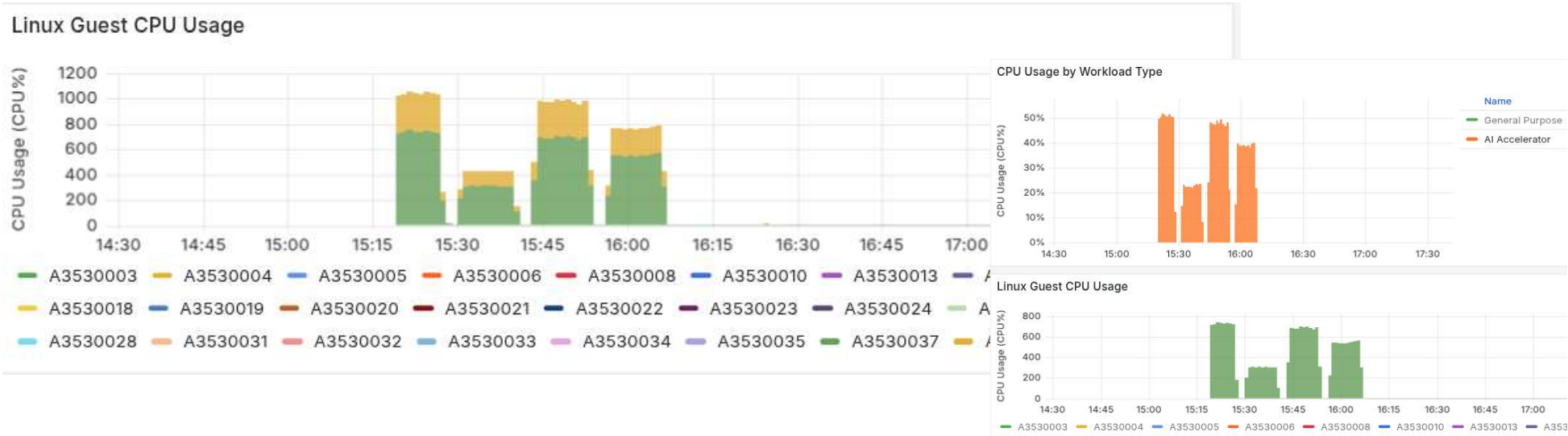
NNPA Instructions Interleaving

NNPA Instructions Interleaving



- The chart shows the number of additional NNPA instructions needed for one complete NNPA operations. An increasing number of additional instructions is related to higher utilization of the AI Accelerator.
- A number above 0% shows that the AI Accelerator is interrupting long-running operations to ensure that all NNPA operations on the AI Accelerator make progress.
- NNPA instructions are issued by the CPU to perform an AI Accelerator function. In some cases, the NNPA instruction ends prematurely (CC3) after a "CPU-determined amount of data processed" in which case the application issues the same NNPA instruction again to resume the operation function where it was suspended. The delay observed by the application shows in NNPA wait time.
- On z17, an NNPA instruction is typically executed on a remote AI Accelerator when the local AI Accelerator is busy. Depending on the configuration of the system, it is not unlikely to see very low number of interleaved NNPA instructions.

Linux Guest CPU Usage



- The chart shows a breakdown of Linux CPU usage by Linux guest. This can be helpful when other non-AI workloads run on the same z/VM system in other Linux guests.
- When studying the performance of an AI workload, it is helpful to select the relevant Linux guests in this chart (via the legend) to see the aggregate CPU usage of these guests in relation to the AI Accelerator usage on the z/VM system and blend out other unrelated usage on z/VM as suggested in the side bar chart.
- Users may want to modify the sample dashboard and select specific parts of the workload to study certain performance aspects of the application.

IBM