3. Performance analysis of MPI applications has been an active area of research. There have been many performance tools developed to support performance MPI applications. Please identify two of these frameworks and compare and contrast the capabilities of the toolsets you have selected. Make sure to cite all your resources. Please do not copy text out of user guides when you discuss the frameworks.

Intel Vtune APS:
Cited from https://docs.nersc.gov/tools/performance/aps/.
APS is a lightweight open source profiling tool for MPI applications, which provides a quick view into a shared memory or MPI application's use of available hardware. It mainly analyze application's time spent in MPI, MPI and OpenMP imbalance, memory access efficiency, FPU usage and I/O and memory footprint. Run the following command to collect data about your MPI application: `nersc$ <mpi launcher> <mpi parameters>aps <my app> <app parameters>.` APS launches the application and runs the data collection. APS is usually used in conjunction with Intel Vtune Amplifier and Intel TAC tools.

Vampir:
Cited from https://slub.qucosa.de/landing-page/?tx_dlf[id]=https%3A%2F%2Fslub.qucosa.de%2Fapi%2Fqucosa%253A703%2Fmets.
Vampir was developed at KFA which now supports the new message passing standard MPI. Vampir translates a given trace file into a variety of graphical views and also it supports an animation mode that can help to locate performance bottlenecks. Sepcifically, Vampir provides single time system snapshots, animation, statistics and time-line system view. Each category is supported by the Vampir environment. Vampir accesses several external tools to perform some of its tasks.

The common aspect is that both methods visualize the appplication and generate various visualization results. They can both evaluate the hardware environment in which MPI operates. However, while APS focuses on the temporal level and evaluates the pre-production environment in which the MPI application is running, vampir focuses on the post-production environment and is more complex to implement.