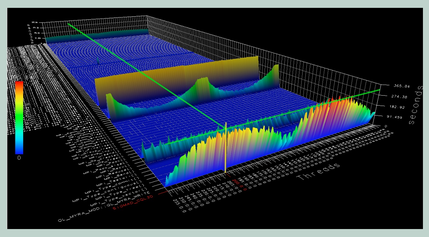
Performance Measurement and Analysis using the TAU Performance System

Allen D. Malony, Sameer Shende, Kevin Huck, Camille Coti, Wyatt Spear, Jeffrey S. Vetter, Mary Hall, Kei Davis,
Ian Foster, Scott Klasky, Kerstin Kleese van Dam, Todd Munson

Funded through ECP 2.3.2.10 (PROTEAS-TUNE) and ECP 2.2.6.08 (CODAR)

What is TAU?



- Tuning and Analysis Utilities (25+ year project)
- http://tau.uoregon.edu
- Comprehensive performance profiling and tracing
  - ✓ Scalable, flexible, portable
  - ✓ Targets all parallel programming/execution paradigms
- Integrated performance toolkit
  - ✓ Instrumentation, measurement, analysis, visualization
  - ✓ Timers, samples, counters, integrated tool callback support, hardware counter support
  - ✓ Widely-ported performance profiling and tracing system
  - ✓ Performance data management and data mining
  - ✓ Open source (BSD-style license)

PROTEAS-TUNE Project Goals

(PROgramming Toolchain for Emerging Architectures and Systems)

Programmer productivity and performance portability are two of the most important challenges facing applications targeting future exascale computing platforms. The PROTEAS-TUNE project is a strategic response to the continuous changes in architectures and hardware (e.g., heterogeneous computing, deep memory hierarchies, nonvolatile memory) that are defining the landscape for emerging ECP systems. PROTEAS-TUNE is a flexible programming framework and integrated toolchain that will provide ECP applications the opportunity to work with programming abstractions and to evaluate solutions that address the exascale programming challenges they face.

Key Capabilities: LLVM; **OpenACC**, **CUDA**, **HIP, OpenCL, OneAPI**; Performance tools with **TAU**; Expertise and software systems for **heterogeneous computing** (GPUs, FPGAs, Manycore) and **deep memory hierarchies** including **nonvolatile memory**; **Performance portability** metrics, tools, and strategies.

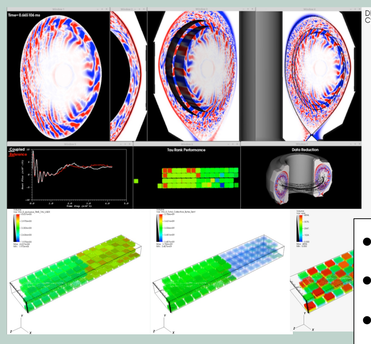
ECP Collaborators : CODAR, ADIOS, WDMApp, CANDLE, NWChemEx, AMReX, Kokkos, others…

2017

- MPI_T added
- Kokkos support
- Native OTF2 tracing
- System monitoring
- ADIOS measurement
- SOS integration
- InteL PIN support
- Plugin system added
  - Tuning plugins
  - Selective measurement
- POSIX I/O metadata capture
- OMPT TR4 prototypes
- LIKWID support
- Caliper integration
- V2.27 released

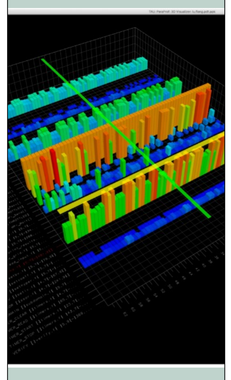
2018

- New, updated plugins, callbacks
- OMPT TR6 extensions
- Additional system monitoring
- MPI_T PVAR tuning
- LIKWID updates
- CUDA 9, 10 updates
- ARM Compiler support
- PerfStubs prototyped
- ROCm support prototyped
- CLACC integration
- Refactored Kokkos support
- Sampling refinements
- V2.28 released



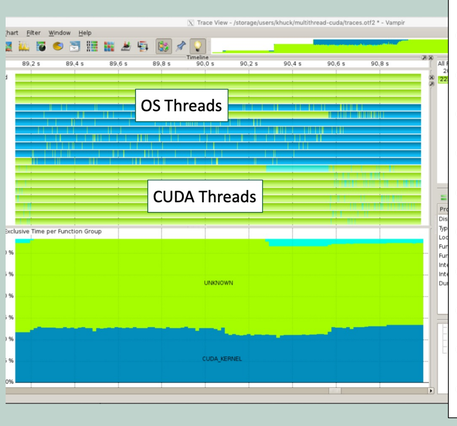
2019

- Python3 + CUDA support
- CUDA/OpenCL enhancements
- OMPT 5.0 update
- F18/Flang instrumentation
- OpenACC updates
- ADIOS2 monitoring output
- Chimbuko integration w/ADIOS2
- V2.29 released



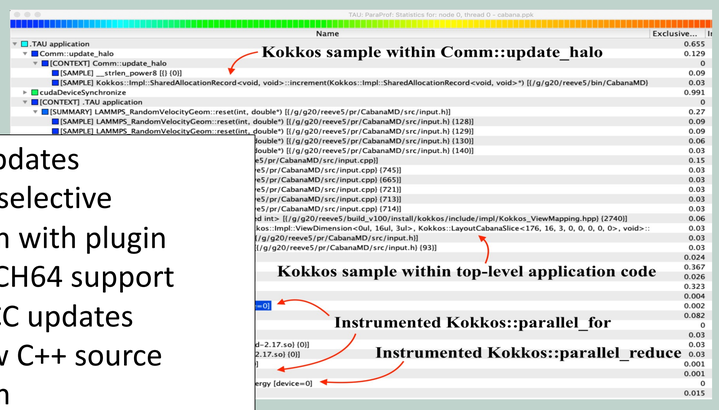
2020

- CUDA, ROCm support updates
- Clang/Clang++ instrumentation plugin
- Enhanced system monitoring
  - NVML
  - PAPI
  - HW/OS state
- Enhanced TensorFlow/PyTorch support
- Intel OneAPI support (Level Zero)
- OMPT 5.1 updates
- CLACC/OpenACC updates
- V2.30 released



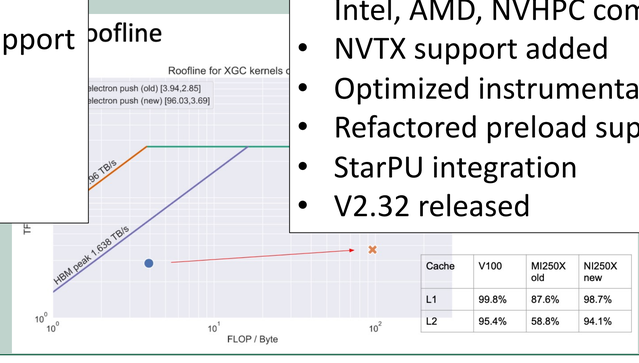
2021

- GPU support updates
- Clang/Clang++ selective instrumentation with plugin
- ARM64FX/AARCH64 support
- CLACC/OpenACC updates
- Prototyped new C++ source instrumentation
- NVHPC compiler support
- AMDClang compiler support
- V2.31 released



2022

- Eliminated pthread limits
- GPU support updates
- Roofline model for AMD MI250X
- Updated OneAPI support
- OMPT 5.2 target offload, test with Intel, AMD, NVHPC compilers
- NVTX support added
- Optimized instrumentation plugin
- Refactored preload support
- StarPU integration
- V2.32 released



Cache	V100	MI250X old	NI250X new
L1	99.8%	87.6%	98.7%
L2	95.4%	58.8%	94.1%

2023

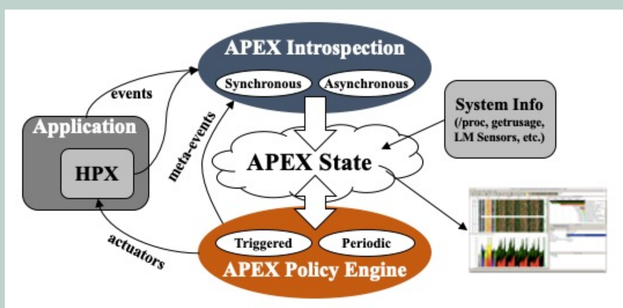
TAU / PROTEAS-TUNE: Final Steps

- **CUDA 11** : Continued updates and testing with A100 architectures and newer
- **OpenMP / OpenACC** : Maintain and update target offload measurement for OpenMP and OpenACC regions executed on target devices.
- **ROCm/HIP** : Continued support for AMD GPUs, HW counters
- **oneAPI** : Update and develop support for latest Pontevecchio GPU
- T**AU compiler wrappers** : enhanced LLVM plugin, source-to-source capabilities

Related Project: APEX

Autonomic Performance Environment for Exascale

https://github.com/UO-OACISS/apex



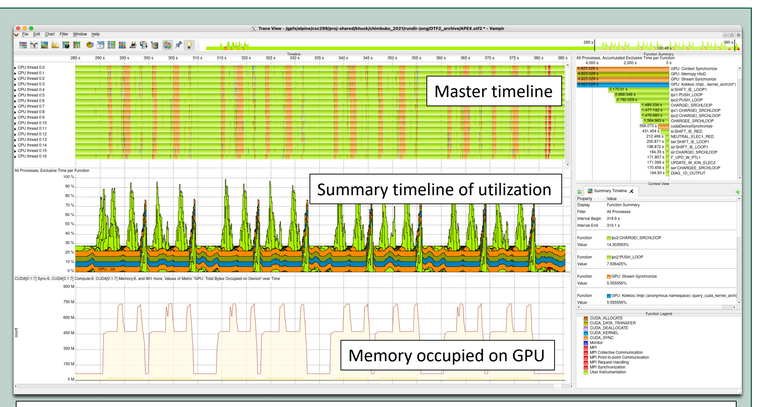


Figure: XGC executed on Summit, measured with APEX, visualized in Vampir, including CUDA and Kokkos activity. APEX supports several asynchronous threading and tasking programming models, including C++ std::async, HPX, OpenMP, OpenACC, Kokkos, Raja, CUDA, HIP. Monitoring support similar to the TAU plugin, providing utilization statistics for the filesystem, network, devices, CPU, GPU, and memory.

This research was supported by the Exascale Computing Project (17-SC-20-SC), a collaborative effort of two U.S. Department of Energy organizations (Office of Science and the National Nuclear Security Administration) responsible for the planning and preparation of a capable exascale ecosystem, including software, applications, hardware, advanced system engineering, and early testbed platforms, in support of the nation's exascale computing imperative.