Automated Performance Analysis with Caliper, SPOT, and Hatchet

Tutorial Introduction

April 12, 2021

David Boehme
Stephanie Brink
Matt LeGendre
Olga Pearce

CASC
Center for Applied Scientific Computing

LLNL-PRES-XXXXXX
This work was performed under the auspices of the U.S. Department of Energy by Lawrence Livermore National Laboratory under contract DE-AC52-07NA27344. Lawrence Livermore National Security, LLC

Lawrence Livermore National Laboratory

Building Automated Performance Analysis Workflows

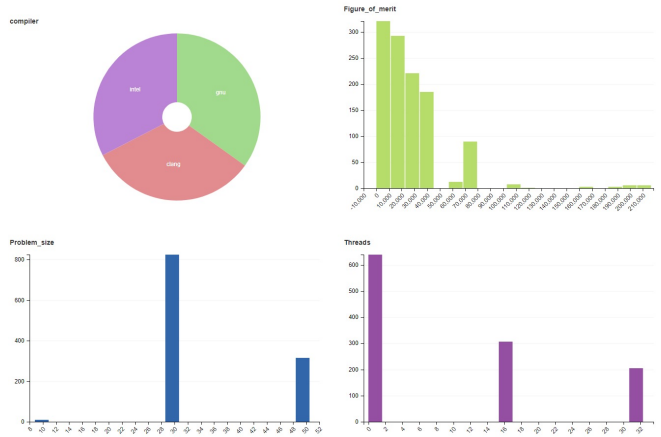
Enabling performance analysis as a *routine, cumulative* activity for HPC software development

Caliper

```
#include <caliper/cali.h>

void LagrangeElements(Domain& domain,
Index_t numElem)
{
   CALI_CXX_MARK_FUNCTION;
// ...
```

Caliper:
Instrumentation and Profiling



SPOT: Analysis of
large collections of runs

```
0.000 foo
├─ 5.000 bar
│  ├─ 5.000 baz
│  └─ 10.000 grault
├─ 0.000 qux
│  └─ 5.000 quux
│     └─ 10.000 corge
│        ├─ 5.000 bar
│        │  ├─ 5.000 baz
│        │  └─ 10.000 grault
│        ├─ 10.000 grault
│        └─ 15.000 garply
└─ 0.000 waldo
```

Hatchet:
Call graph analysis in Python

Tutorial Outline

Time (ET)		Presenter
2:35 - 3:30	The Caliper Performance Profiling Library	David Boehme, LLNL
3:30 - 3:45	15-minute Break	
3:45 - 4:45	Analyzing Large Collections of runs in SPOT (with hands-on exercises)	Matt LeGendre, LLNL
4:45 - 5:00	15-minute Break	
5:00 - 6:00	Call-graph Analysis with the Hatchet Python Library (with hands-on exercises)	Olga Pearce, LLNL

Links

- Caliper: https://github.com/LLNL/Caliper
- Caliper Documentation: https://llnl.github.io/Caliper
- Hatchet: https://github.com/hatchet/hatchet
- Hatchet Documentation: https://hatchet.readthedocs.io
- SPOT: https://github.com/LLNL/spot2_container