Development of a new C++ Performance API (PAPI++) software package from the ground up that offers a standard interface and methodology for using low-level performance counters in CPUs, GPUs, on/off-chip memory, interconnects, I/O system, and energy/power management. PAPI++ is building upon classic PAPI functionality and strengthens its path to exascale with a more efficient and flexible software design, one that takes advantage of C++’s object-oriented nature but preserves the low-overhead monitoring of performance counters and adds a vast testing suite.

**ECP SCOPE**
Exa-PAPI++ is preparing PAPI support to stand up to the challenges posed by exascale systems by:

**GOAL 1** Widening its applicability and providing robust support for exascale hardware resources.

**GOAL 2** Supporting finer-grain measurement and control of power, thus offering software developers a basic building block for dynamic application optimization under power constraints.

**GOAL 3** Extending PAPI to support Software-Defined Events that originate from the ECP software stack and are treated as black boxes (e.g. communication and math libraries, runtime systems, etc.).

**GOAL 4** Applying semantic analysis to hardware counters so that the application developer can better make sense of the ever-growing list of raw hardware performance events.

The team will be channeling the monitoring capabilities of hardware counters, power usage, software-defined events into a robust PAPI++ software package. PAPI++ is meant to be PAPI’s replacement—with a more flexible and sustainable software design.

**PERFORMANCE COUNTER MONITORING CAPABILITIES**

**SUPPORTED ARCHITECTURES**
- AMD GPUs
- NVIDIA GPUs
- Intel Xeon processors
- Intel GPUs
- IBM Power
- ARM processors
- Cray CPUs
- Eka-Go37 Power+DM
- ARM Virtex
- KVRX Virtex
- VMWare virtual environment

**SUPPORT FOR GPUs: AMD and NVIDIA**
Activity 1: Performance counter monitoring
- Develop support for NVIDIA monitoring capabilities for GPUs on Summit.
  - Added PAPI capabilities for monitoring TESLA V100 + NVLINK
- Develop support for AMD GPUs Monitoring Capabilities:
  - Development of a new PAPI ROCm component

Activity 2: Power monitoring and capping support
- Develop support for NVIDIA power management capabilities for GPUs on Summit.
  - Added PAPI capabilities for monitoring TESLA V100:
    - Power consumption and power capping support
    - Fan speed, temperature
- Develop support for AMD GPUs power monitoring:
  - Development of a new PAPI ROCm-smi component

**POWER AWARENESS EXAMPLE**
- Power Reading and Capping with PAPI on TESLA V100 GPUs
- Enables developers to change run profiles to reduce energy cost

**SOFTWARE-DEFINED EVENTS IN PAPI**
**KEY POINTS ABOUT SDEs**
- New measurement possibilities:
  - Tasks stolen, matrix residuals, partial results reached, arguments passed to functions
- Any tool can read PAPI SDEs:
  - SDEs from a library can be read with `PAPI_start()`/`PAPI_stop()`/`PAPI_read()`.
- Low overhead:
  - Performance critical codes can implement SDEs with zero overhead.
- Easy to use, with rich feature set:
  - Pull-mode & push-mode, read-write counters, sampling/overflowing, counters, groups, recordings, statistics, thread safety, custom callbacks

**DIFFERENT ACCESS MODELS**

**Pull mode: Zero overhead**

```
void PAPI_read()
```

```
X++;
X=X7;
```

**Application**

**PAPI**

**Library of SDEs**

```
create_counter();
callback();
```

**Push mode: Determinism and precision**

```
void *papi_sde_init(char *lib_name, int event_count);
void *papi_sde_register_counter(void *handle, char *event_name, int type, int mode, void *counter);
```

```
void *papi_sde_describe_counter(void *handle, char *event_name, char *event_description);
```

**ECP PROJECTS AND 3RD PARTY TOOLS APPLYING PAPI**

<table>
<thead>
<tr>
<th>ECP DTE (PARSEC)</th>
<th>ECP LLNL-ATDM (Caliper)</th>
<th>ECP SNL-ATDM (Kokkos)</th>
<th>ECP Proteas (TAU)</th>
</tr>
</thead>
</table>

**ECP HPC Toolkit (HPCToolkit)**

```
Score-P
```

```
http://www.score-p.org
```

**Vampir**

```
Tu Dresden
```

```
http://www.vampir.eu/
```

**Scalasca**

```
FZ Juelich, TU Darmstadt
```

```
http://scalasca.org/
```

**PerfSuite**

```
http://perfsuite.ncsa.uiuc.edu/
```

**OpenSpeedshop**

```
http://openspeedshop.org/
```

**SyPablo**

```
www.renci.org/research/pablo
```

**ompP**

```
http://www.openspeedshop.org/
```

**SUPPORTED SOFTWARE AVAILABLE AT**
http://icl.utk.edu/exa-papi/

**SPONSORED BY**

[ICL Innovative Computing Laboratory](https://icl.utk.edu)
[University of Tennessee](https://www.utk.edu)
[Energy](https://energy.gov)
[Office of Science](https://science.energy.gov)
[NSF](https://www.nsf.gov)
[Exascale Computing Project](https://www.esa.europa.eu)