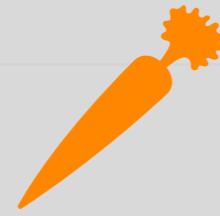


sys-sage: A Fresh View on Dynamic Topologies & Attributes of HPC Systems



Stepan Vanecek
stepan.vanecek@tum.de
TU Munich

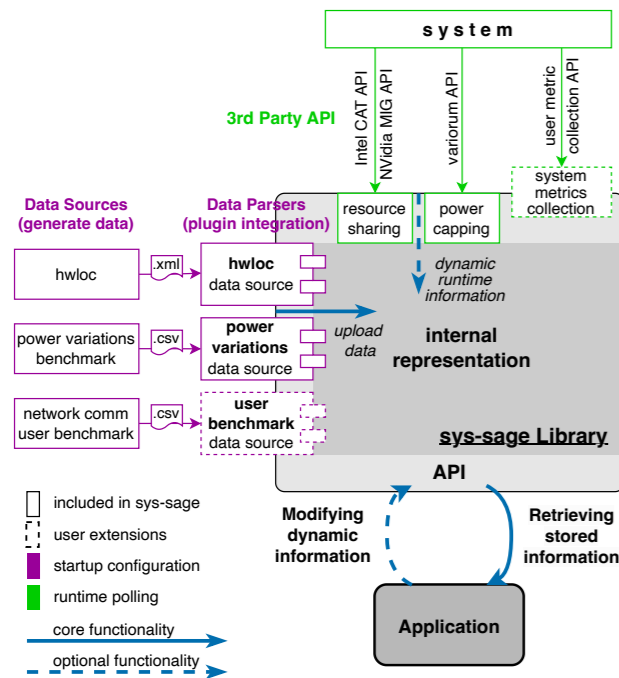
Martin Schulz
schulzm@in.tum.de
TU Munich

What is sys-sage?

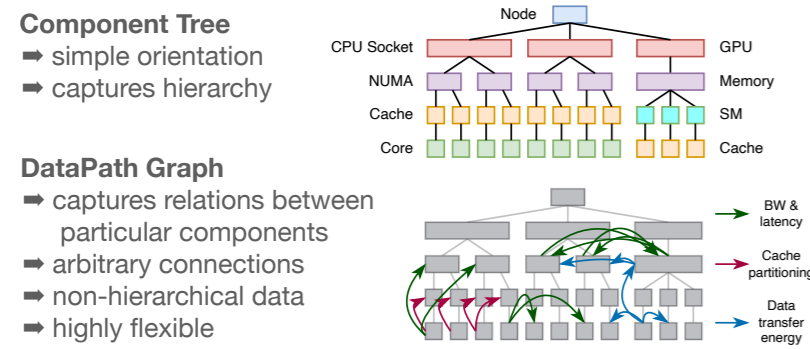
sys-sage is a user-side library focussing on collection, storage, and provision of arbitrary HW-relevant information within an HPC system.

sys-sage provides the needed context between the plethora of applications, tools, and benchmarks providing some information describing the increasingly complex modern HPC systems. It manages the context for both static and dynamic data, which become available at different stages of the application's lifecycle.

Internal Architecture

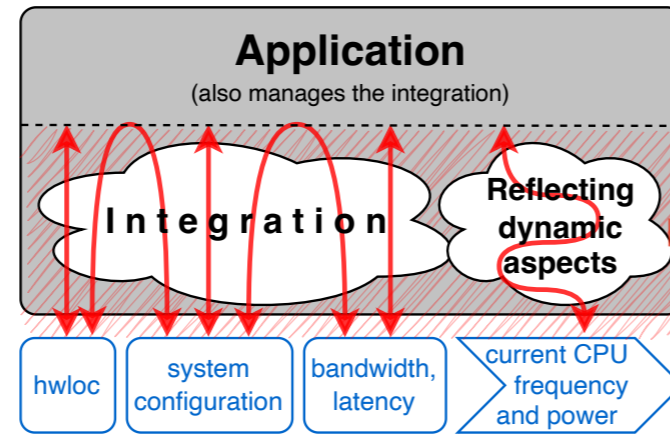


Internally built on two complementary structures



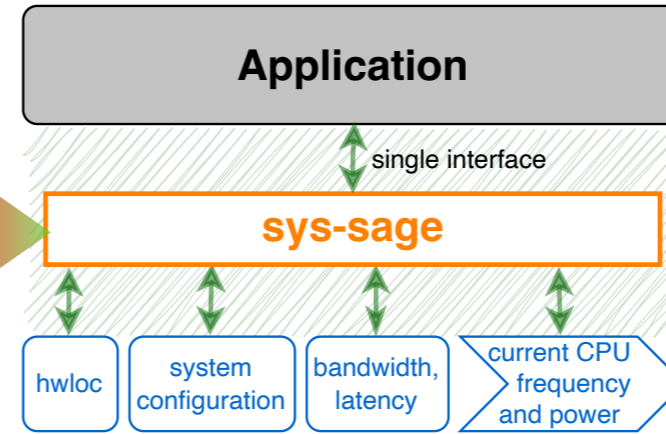
Why use sys-sage?

Traditional approach



- Integrating many tools with different APIs
- Connecting and correlating the data from many different backends not feasible
- Storing only specific provided data
- Usually only either static or dynamic information
- Forcing a deep integration of chosen tool(s)
 - ➔ no flexibility or adaptability

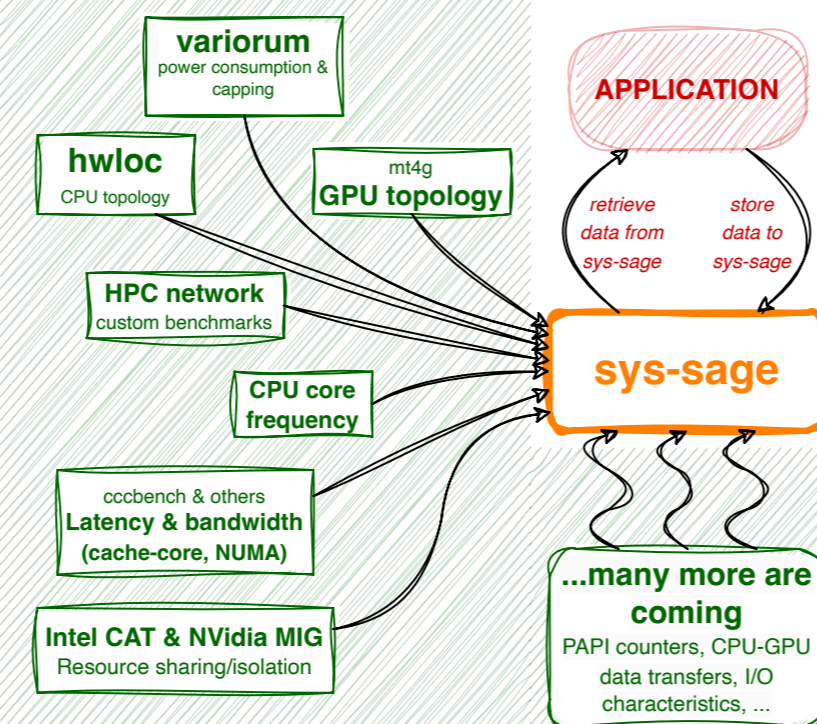
sys-sage approach



- + No redundant backends integration
- + Automatically connecting and correlating the information
- + Maintaining arbitrary data
- + Capturing both static and dynamic information
- + Offering unique and universal structure to capture HW systems and their attributes

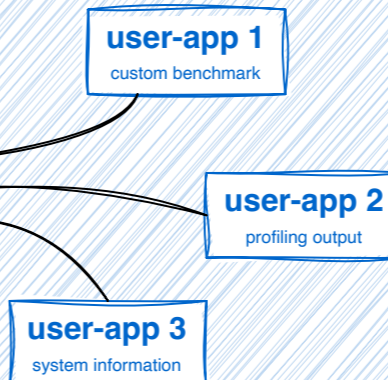
Providing data to sys-sage

Integration out-of-the-box



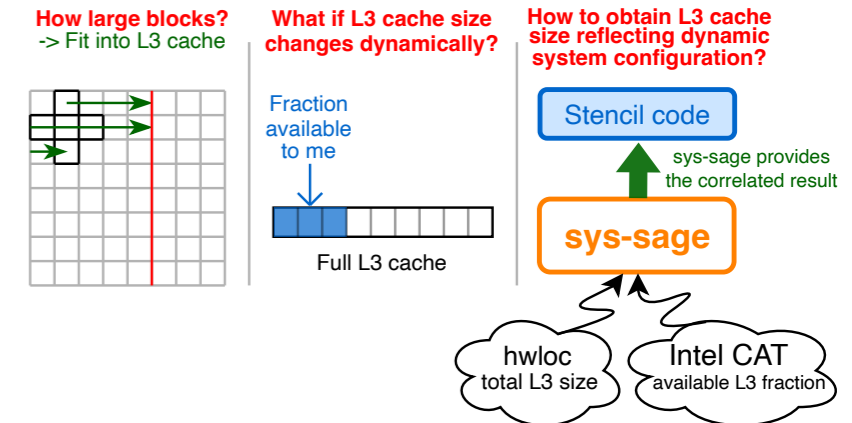
User-defined integration

ANY ADDITIONAL TOOL OR INFORMATION CAN EASILY BE INTEGRATED BY THE USER

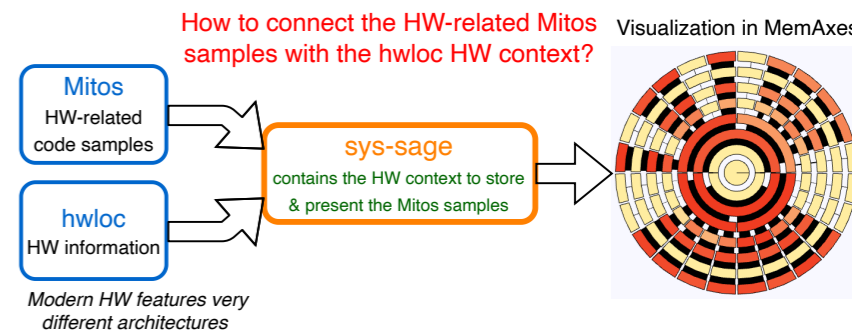


Using sys-sage

Cache-aware algorithms on systems with dynamically changing system properties



Sample-based performance visualization



General areas of usage:

- ✓ Code tuning
- ✓ Performance monitoring & visualization
- ✓ Performance estimation
- ✓ Performance/system modelling & simulations
- ✓ Scheduling (node- / thread-based /...)
- ✓ Co-scheduling
- ✓ Power-management
- ✓ System design

Try sys-sage out and get in touch with us!

<https://github.com/caps-tum/sys-sage>

stepan.vanecek@tum.de

spack install sys-sage

