SEANERGYS: Integrating Monitoring, Analytics, and Resource Management for Energy-efficient HPC System Operations

Invited Paper

Estela Suarez
e.suarez@fz-juelich.de
Jülich Supercomputing Centre,
Forschungszentrum Jülich GmbH
Jülich, Germany
University of Bonn
Bonn, Germany

Andrea Borghesi
andrea.borghesi3@unibo.it
University of Bologna
Bologna, Italy

Sonja Happ
sonja.happ@par-tec.com
ParTec AG
Munich, Germany

Hans-Christian Hoppe
h.hoppe@fz-juelich.de
Jülich Supercomputing Centre
Forschungszentrum Jülich GmbH
Jülich, Germany

Michael Ott
ott@lrz.de
Leibniz Supercomputing Centre
Garching, Germany

Simon Pickartz
pickartz@par-tec.com
ParTec AG
Munich, Germany

Martin Schulz
schulzm@in.tum.de
Technical University of Munich
Munich, Germany

Hannes Tröpgen
hannes.troepgen@tu-dresden.de
ZIH, CIDS, TU Dresden
Dresden, Germany

Abstract

SEANERGYS is an EuroHPC JU project developing an integrated software suite to optimize the operation of HPC systems, aiming to reduce energy consumption across real-world workload mixes. Additional objectives include optimizing resource utilization, enhancing system throughput, and minimizing response times.

This paper introduces the overall SEANERGYS software architecture, covering its three main elements (monitoring, AI data analytics, and dynamic scheduling and resource management) leveraging a central data plane to connect them.

CCS Concepts

• **Computer systems organization** → **Power and energy**; **Parallel and distributed computing**; *System management*; *Performance monitoring*; • **Software and its engineering** → *System software*.

Keywords

HPC, energy-efficiency, resource management, system monitoring, operational data analytics, system software

ACM Reference Format:
Estela Suarez, Andrea Borghesi, Sonja Happ, Hans-Christian Hoppe, Michael Ott, Simon Pickartz, Martin Schulz, and Hannes Tröpgen. 2026. SEANERGYS: Integrating Monitoring, Analytics, and Resource Management for Energy-efficient HPC System Operations: Invited Paper. In *Proceedings of the 23rd ACM International Conference on Computing Frontiers (CF Companion '26), May 19–21, 2026, Catania, Italy*. ACM, New York, NY, USA, 4 pages. https://doi.org/10.1145/3801488.3806245

ACM ISBN 979-8-4007-2569-2/2026/05
https://doi.org/10.1145/3801488.3806245

1 Introduction

Increasing energy and power demands are key challenges we must address as we scale High-Performance Computing (HPC) and AI systems to meet the ever-growing computational demands of industry and research users [8]. Consequently, there is a growing need for new solutions that enable HPC centers to reduce energy consumption, enhance resource utilization, and optimize workloads.

Significant efforts have been exerted on the development of more energy-efficient Hardware (HW): new architectures [9], processors with higher core counts and vectorization [1, 2], accelerators for critical workloads, and optimized data center infrastructure and cooling [8]. However, HW developments alone are insufficient, as they are naturally limited in scope, cannot act in a globally coordinated fashion, and lack mechanisms to influence user behavior. They have to be complemented with matching energy-aware Software (SW) solutions. Proof-of-concept implementations of such approaches have demonstrated potential [3, 4, 7, 10, 11], but their isolated deployment, early state, and limited maturity have so far hindered widespread adoption in production HPC centers.

The SEANERGYS project integrates monitoring, data analysis, and resource management capabilities into a unified, production-ready SW infrastructure that works hand-in-hand with matching HW developments and interacts with users, providing needed optimization data. Key project contributions include:

- **Comprehensive Monitoring Infrastructure (CMI)**: a unified holistic monitoring system for HPC system hardware, system software, applications, and supporting infrastructure.
- **Artificial Intelligence Data Analytics System (AIDAS)**: a data analytics framework extracting insights from monitoring data, delivering accurate performance, energy, and power predictions to guide runtime decisions.
- **Dynamic Scheduling and Resource Management System (DSRM)**: an integrated set of enhanced resource managers that enhance system utilization, throughput, response times, and energy efficiency via policy-driven, multi-objective, and dynamic scheduling.

All the above are delivered as **high-quality, deployable software solutions** at advanced Technology Readiness Levels (TRLs), suitable for operational use in HPC centers.

2 Requirements and Use Cases

The SEANERGYS suite is a comprehensive SW solution to improve throughput, energy efficiency, and resource utilization in HPC systems. It fulfills the needs of all key stakeholders: end-users, application and tool developers, sysadmins, and technical support teams. Requirements are described via *Use Cases (UCs)*, leading to a catalog of twelve UCs (see Figure 1) that cover the required functionalities from the SEANERGYS SW suite: monitoring, alerting, predicting, optimizing, and scheduling.

A Use Case (UC) is an end-to-end scenario that delivers value to one or more stakeholders by composing data, models, mechanisms, policies, and —where applicable— control loops. The implementation of a UC will utilize multiple SEANERGYS SW components to achieve its objective. Some components serve multiple UCs, and some UCs depend on the outputs of others (e.g., predictions feeding scheduling). Figure 1 visualizes the role (*observing* vs. *acting*) and scope (*user*- vs. *system-level*) of all UCs, and clusters them according to their main field of responsibility: *monitoring/observability* (green) focuses on telemetry acquisition, anomaly detection, and visual analytics; *prediction and intelligence* (red) provides reusable predictive models that serve as inputs to optimization and scheduling mechanisms; *energy and power optimization* (yellow) enforces infrastructure limits and optimizes energy-to-solution and carbon footprint; and *advanced scheduling* (blue) extends traditional HPC scheduling with adaptive and policy-driven functionalities.

Sysadmins get insight into the operational status of the HPC system and receive timely notifications through a monitoring system with alerting capabilities (UC1), presented through customizable and interactive dashboards (UC2), and obtain insights on software library usage (UC12) so they can optimize runtime and workflow paths and deployment choices. This picture is enriched with an Machine Learning Operations (MLOps)-based modeling service (UC5) that characterizes jobs and system status, and that provides predictions to upstream decision points.

Operations are driven by a set of complementary modes:

- **Power Capping**: guarantees operation under a specified cap while optimizing application performance (UC3);
- **Energy-efficiency**: promotes energy as a first–class objective and seeks the best energy-performance operating point (UC6);

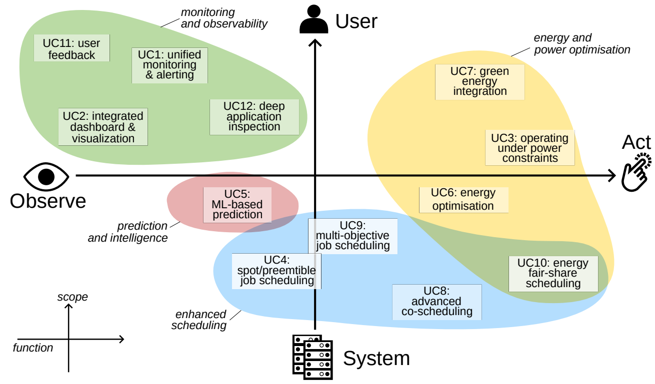

Figure 1: Use Cases (UCs) mapping with respect to their function (*observing* to *acting*) and scope (*system-* to *user-level*). The colored blobs group the UCs into *clusters* based on their primary field of responsibility (labeled in italics).

- **Carbon-aware**: minimizes Carbon Dioxide Equivalent (CO_2e) emissions by aligning work with low–carbon windows (UC7);
- **Site-level performance optimization**: exploits advanced mechanisms, such as co-scheduling (UC8), and spot jobs to raise throughput and responsiveness (UC4).
- **Multi-objective**: allows site administrators to define and enforce weighted combinations of different targets according to policy and context (UC9).

Application developers and end-users are included in the optimization loops: energy becomes part of fair share resource allocation (UC10) to incentive an alignment with center objectives. Actionable feedback reports (UC11) help users and developers achieve high resource utilization and expose job misconfigurations. This can improve both turnaround time and efficiency.

3 Software Architecture

The SW architecture of the *SEANERGYS suite* depicted in Figure 2 has been designed to support all UCs documented in Section 2. The CMI (Section 3.1) gathers data from HW and SW sensors, and correlates it with scheduler information to provide job- and system-based telemetry. The AIDAS (Section 3.2) leverages AI models trained with a vast set of operational data of the participating HPC sites. It fingerprints resource usage patterns, predicts future job behavior, and identifies complementary job profiles for potential co-scheduling. Users can receive feedback on energy and resource use for each job, plus suggestions for improvements. Finally, the DSRM (Section 3.3) utilizes these insights to develop scheduling policies that maximizes resource utilization and energy efficiency, and supports jobs/applications with dynamic and adaptable resource profiles. These three main SEANERGYS blocks are interconnected and exchange data via the data plane (Section 3.4).

Modularity is a key design principle: HPC sites may substitute specific SW components by local solutions, provided that these are adapted to use the data plane Application Programming Interfaces (APIs). Replacing certain central components —like the system

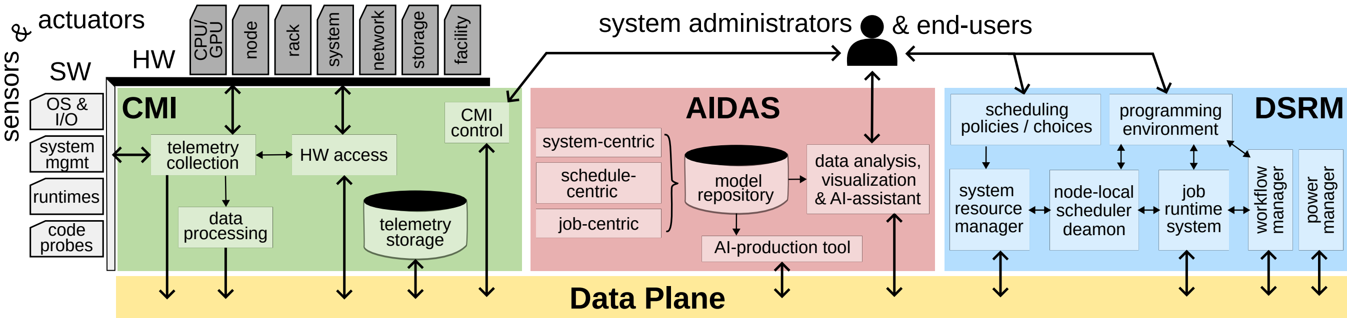

Figure 2: SEANERGYS architecture. High-level view including the main elements of the Comprehensive Monitoring Infrastructure (CMI), Artificial Intelligence Data Analytics System (AIDAS) and Dynamic Scheduling and Resource Management System (DSRM). All are connected via the data plane.

resource manager and scheduler— might, however, lead to advanced SEANERGYS features being lost.

3.1 CMI: Comprehensive Monitoring infrastructure

Traditional monitoring focuses on the health of HW and core services. However, HPC centers striving for energy efficiency require broader monitoring, including energy flows, job efficiency, runtime behavior, application and Input Output (I/O) patterns, power consumption, and infrastructure, such as cooling and power distribution. Advanced solutions exist at some sites, but incompatible naming, data organization, and metadata hinder SW portability and data sharing, causing fragmentation and duplicated effort.

SEANERGYS is defining a *monitoring standard* covering four domains: system HW, system SW, applications, and supporting infrastructure. It specifies interfaces and protocols for telemetry ingestion and queries, alongside a unified naming scheme and ontology. Thus, it provides a common foundation for all SEANERGYS components and facilitates deployment at different sites, reducing maintenance costs. The project will deliver a reference implementation of the CMI, leveraging existing open source components, where possible, and focusing on integrating diverse data acquisition mechanisms into collection daemons. The result will be a unified solution that is deployed across all partner HPC centers and which incorporates advanced features using the AIDAS models.

The green box in Figure 2 shows the main CMI components. The *telemetry collection* component queries *sensor* readings or *actuator* states via either the *HW access* daemon or third-party APIs. It also publishes telemetry to the data plane, possibly after some *data processing*, where it is accessible to subscribers. The *HW access* component configures sensors, actuators, and HW counters, provides access to other components via the data plane, and arbitrates conflicting requests. Monitoring data is stored in the *telemetry storage* and made available via database queries. Control commands pass through the data plane to connected components. Typically, nodes in the system and facility host a HW access and a telemetry collection daemon, with management nodes handling the telemetry storage. System administrators can deploy configurations and start/stop services via the *CMI control*.

As monitoring data may contain confidential (e.g., vendor-specific) or personal data, SEANERGYS will define anonymization policies, and other mechanisms/tools to allow sharing data with the larger community and enable future research with realistic datasets.

3.2 AIDAS: Artificial Intelligence and Data Analytics System

The AIDAS part, depicted in the red box in Figure 2, turns the CMI monitoring data into insights, predictions, and actionable feedback. It has three main components.

The *model repository* is a set of curated models trained with collected monitoring data that perform HW/system-centric, schedule-centric, and job-centric analysis/prediction. The model repository will be compatible with different datasets and enable consistent evaluation, quality assurance, traceability, and reproducibility.

The *AI-production tool* supports the full MLOps life cycle, including training/retraining, model evaluation, model deployment, and online inference. It interfaces the model repository with the rest of the architecture, injects data to the data plane and runs a decision-support and prediction service on the operational platform, directly supporting UC5 in Figure 1 and providing predictions to operational components, primarily the DSRM.

The *data analysis, visualization and AI assistant* provides an interactive, web-based dashboard accessible to all stakeholders, in particular end-users and system administrators. Its chatbot-driven interface and dashboard templates support common use cases, for instance detailed job reports for post-mortem analysis as well as fine-grained reporting, based on metrics obtained from CMI and DSRM via the data plane.

3.3 DSRM: Dynamic Scheduling and Resource Management

The DSRM part (blue box in Figure 2), consists of managers that implement scheduling and resource management decisions to optimize operations based on the UC definitions. This includes maximizing system utilization and energy efficiency. DSRM operation is driven by CMI data and information from AIDAS. A *multi-objective scheduler* (part of the *system resource manager*) makes scheduling and resource allocation decisions by balancing site-specific *scheduling policies*.

System administrators define *scheduling policies* and end-users adjust their *scheduling choices* in job submission scripts. Both are handled by the *system resource manager*, which is based on Flux [6]. The *system resource manager*, the *job runtime system*, and the *node-local scheduler daemon* are Flux *brokers* that communicate directly using the Flux Remote Procedure Call (RPC) protocol (e.g., to get or set information about jobs and resource changes).

The *node-local scheduler daemon* performs resource allocation on each compute node, and propagates information to the *job runtime system* and application processes by the Process Management Interface for Exascale (PMIx) API or the SEANERGYS data plane.

The *job runtime system* exchanges information with the *workflow manager* via the data plane to ensure meeting global job constraints and job parameter coordination. The *workflow manager* interacts with the *system resource manager* via the data plane (e.g., to set resource constraints, get information about job and queue status, or to request job submission). The *programming environment* employed by the user applications interacts with all these components to ensure proper resource allocation and communication between processes running on different nodes.

Finally, the *power manager* implements power-saving strategies, such as power capping, according to site policies. It retrieves information from other SEANERGYS components and configures power consumption limit of the respective HPC center (via the DSRM's *HW access* using the data plane capabilities, and shares information in the same way.

3.4 Data Plane

The data plane (yellow box in Figure 2) is the central communication infrastructure of SEANERGYS. It provides a coherent interface to all components and enables interoperability and portability.

The data plane supports two communication modes: **Publish/Subscribe (Pub/Sub) communication** is intended for $1:n$ scenarios where a *publisher* sends data to one or more *subscribers*. Data is organized in *topics*, to which consumers can subscribe, and a central broker distributes messages. Publishers and subscribers remain decoupled, as neither side needs direct knowledge of the other. This mode is primarily intended for monitoring data, e.g., telemetry collected within CMI and consumed by AIDAS or DSRM.

Request/Response (Req/Res) communication is typically used for $1:1$ client–server interactions, and also supports $1:n$ (broadcast) or $n:1$ (reductions) communication patterns. One or more *clients* explicitly address one or more *servers*, unlike with Publish/Subscribe (Pub/Sub), where communication groups are defined by topics. This mode is used , e.g., for database queries, calls to AIDAS services, or to drive HW settings.

All accesses to the data plane are performed via a single API, which provides the core functions to perform communication with Request/Response (Req/Res) and Pub/Sub semantics. To serialize and de-serialize data sent through the data plane, Cap'n Proto [5] has been chosen.

A *locator service* within the data plane enables components to discover communication partners for Req/Res and retrieve their endpoints. For Pub/Sub communication, brokers deployed by SEANERGYS handle message routing.

By default, all communication in SEANERGYS uses the data plane. However, exceptions are made if that is infeasible, like for components developed outside of the project (e.g., Flux using its internal protocol between brokers) and in situations where ultra-low latency is required (e.g., for load balancing between applications on the same node).

4 Conclusions and Next Steps

The SEANERGYS project addresses the growing energy and resource challenges of HPC by developing a unified, production-ready SW suite for energy-aware operation. Combining holistic monitoring (CMI), predictive analytics (AIDAS), and dynamic resource management (DSRM), it aims at improving energy efficiency, resource utilization, and workload steering, providing actionable insights for operational decisions and job/application optimizations.

Within the first months of the project, the Use Cases (UCs) and requirements have been defined, the SW architecture established, and key decisions taken, including selecting the Machine Learning (ML) models for the AIDAS *model repository*, initial data set collection, and the use of Flux as the system-level scheduler.

After this initial phase, the SEANERGYS project will develop the SW components, tests them, and proceed to validation on multiple European HPC centers with diverse HW architectures. If successful, it will continue to deployment on petascale and exascale systems. These efforts will ultimately help us advance operational efficiency in next-generation HPC infrastructures.

Acknowledgments

The authors acknowledge the contribution of the whole SEANERGYS consortium to the ideas and work presented here. SEANERGYS receives funding from the EuroHPC JU under grant agreement 101177590. The JU receives support from the European Union's Horizon European Research and (for this project) from the Innovation Programme and Czech Republic, France, Germany, Greece, Italy, Luxembourg, and Spain. LLM tools were used for language editing and condensation. All ideas, interpretations, and conclusions remain solely those of the authors.

References

[1] DARE Consortium. 2026. *DARE project.* Retrieved March 26, 2026 from https://dare-riscv.eu/
[2] EPI Consortium. 2026. *European Processor Intitiative.* Retrieved March 26, 2026 from https://www.european-processor-initiative.eu/
[3] REGALE consortium. 2026. *The REGALE project.* Retrieved March 26, 2026 from https://regale-project.eu/
[4] Julita Corbalan, Lluis Alonso, Jordi Aneas, and Luigi Brochard. 2020. Energy Optimization and Analysis with EAR. In *2020 IEEE International Conference on Cluster Computing (CLUSTER)*. 464–472. doi:10.1109/CLUSTER49012.2020.00067
[5] Cap'n Proto development team. 2026. *Cap'n Proto.* https://capnproto.org
[6] Flux development team. 2026. *Flux Framework.* https://flux-framework.org/
[7] Ahmad Tarraf et al. 2024. Malleability in Modern HPC Systems: Current Experiences, Challenges, and Future Opportunities. *IEEE Transactions on Parallel and Distributed Systems* 35, 9 (2024), 1551–1564. doi:10.1109/TPDS.2024.3406764
[8] Estela Suarez et al. 2025. Energy-aware operation of HPC systems in Germany. *Frontiers in High Performance Computing* Volume 3 - 2025 (2025). doi:10.3389/fhpcp.2025.1520207
[9] Ivan Ratković et. al. 2015. Chapter One - An Overview of Architecture-Level Power- and Energy-Efficient Design Techniques. Advances in Computers, Vol. 98. Elsevier, 1–57. doi:10.1016/bs.adcom.2015.04.001
[10] Alessio Netti, Micha Müller, Axel Auweter, Carla Guillen, Michael Ott, Daniele Tafani, and Martin Schulz. 2019. From facility to application sensor data: modular, continuous and holistic monitoring with DCDB. In *Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis* (Denver, Colorado) *(SC '19)*. Association for Computing Machinery, New York, NY, USA, Article 64, 27 pages. doi:10.1145/3295500.3356191
[11] DEEP projects consortia. 2026. *The DEEP projects.* Retrieved March 26, 2026 from https://deep-projects.eu/