

PROGRESS IN THE DEVELOPMENT OF THE COMMUNITY PARTICLE ACCELERATOR LANGUAGE STANDARD (PALS)

D. Sagan¹, Z. Brencic², A. Edelen³, R. Roussel³, T. Hellert⁴, A. Huebl⁴, C. Mitchell⁴, J. Qiang⁴, E. Zoni⁴, J.-L. Vay⁴, D. Kallendorf⁵, D. Winklehner⁶, D.L. Bruhwiler⁷, E. Valetov^{8,16}, T. Satogata⁹, C. Tennant⁹, H. Zhang⁹, K. Brown¹⁰, W. Lin¹⁰, L. Deniau¹¹, N. Kuklev¹², Y. El Hayek¹³, Y. Hao¹⁴, A. Brynes¹⁵

¹Cornell University (CLASSE), Ithaca, NY, USA

²Jožef Stefan Institute, Ljubljana, Slovenia

³SLAC National Accelerator Laboratory, Menlo Park, CA, USA

⁴Lawrence Berkeley National Laboratory, Berkeley, CA, USA

⁵Technical University of Darmstadt, Darmstadt, Germany

⁶Massachusetts Institute of Technology, Cambridge, MA, USA

⁷RadiaSoft LLC, Boulder, CO, USA

⁸Michigan State University, East Lansing, MI, USA

⁹Thomas Jefferson National Accelerator Facility, Newport News, VA, USA

¹⁰Brookhaven National Laboratory, Upton, NY, USA

¹¹European Organization for Nuclear Research, Geneva, Switzerland

¹²Fermi National Accelerator Laboratory, Batavia, IL, USA

¹³GSI Helmholtz Center for Heavy Ion Research, Darmstadt, Germany

¹⁴Facility for Rare Isotope Beams, East Lansing, MI, USA

¹⁵Science and Technology Facilities Council, UK

¹⁶University of Hawai'i at Mānoa, Honolulu, HI, USA

Abstract

The Particle Accelerator Language Standard (PALS), originally named the Particle Accelerator Lattice Standard, is a community effort to create an open standard to promote lattice and other information exchange for particle accelerators. PALS development is a community-wide international effort involving accelerator physicists from multiple institutions. While it started primarily as a lattice standard for beam dynamics simulations, PALS is now being extended to support other particle accelerator activities, in particular, accelerator operation.

With new accelerators that are becoming more complex, and with larger collaborations and the increasing imprint of artificial intelligence in all accelerator activities (from design to operation to workforce development), the imperative for a common, standardized accelerator ontology has been transitioning from “nice-to-have” to “must-have”.

Presented here is the status of PALS and its relationship to other projects. This includes two particle accelerator related projects within the US DOE Genesis Mission: the Multi-Office Accelerator Team (MOAT) project and the Nuclear Physics AI-Ready Accelerator Data (NARAD) project.

INTRODUCTION

Modern accelerator facilities depend on shared software for design, simulation, commissioning, operations, optimization, and analysis. As machines grow more complex, collaborations more distributed, and AI tools become more

relevant, a persistent interoperability problem has become more urgent: accelerator information is often encoded in facility-specific lattice files, databases, controls middle layers, process-variable names, archivers, and operational conventions. Consequently, tools developed for one machine are difficult to reuse at another without substantial translation and local expertise.

The Particle Accelerator Language Standard (PALS) is a community effort to develop a common, open representation for accelerator information. PALS began as the Particle Accelerator Lattice Standard, focused on lattice exchange for beam-dynamics simulation and analysis. That scope remains essential, but cross-facility workflows now require a broader language describing accelerator objects, hierarchy, roles, parameters, capabilities, signals, constraints, and links to facility-specific controls and data systems. This expanded scope aligns with Nuclear Physics AI-Ready Accelerator Data (NARAD) development and Multi-Office Accelerator Team (MOAT) [1] needs for portable, machine-readable accelerator context for AI-ready data products and reusable agentic workflows.

FROM “LATTICE” TO “LANGUAGE”: AN EXPANDING SCOPE

PALS began primarily as a lattice standard [2]: a common way to describe beamline elements, their parameters, and their organization into lines and machine complexes for simulation, exchange, and analysis. That scope remains

essential, but it is no longer sufficient for the emerging cross-facility AI ecosystem. This need is particularly visible in emerging accelerator efforts. The NARAD project focuses on AI-ready accelerator data infrastructure, while the MOAT project focuses on shared AI tools for accelerator design and operations. Both require software agents, optimization workflows, and data services to reason across facilities whose control systems, channel names, control system abstraction (“middle layers”), and operational conventions differ substantially. In that setting, a standard must describe not only where elements are in a lattice, but also what those elements are, what they can control or observe, what signals implement those capabilities, and how those signals are bound to facility-specific data streams.

This motivates the change in name from the “Particle Accelerator *Lattice* Standard” to the “Particle Accelerator *Language* Standard”. Here, “language” denotes a shared accelerator vocabulary and grammar for expressing machine structure, device roles, signals, parameters, constraints, and relationships. In knowledge-representation terms, this is an accelerator ontology: a machine-readable model that names the important classes of things in the domain and states how they are related. Its purpose is not to replace local naming conventions, databases, or middle layers, but to provide a semantic layer through which tools can interpret local objects consistently [3]. In this expanded scope, the lattice remains the organizing backbone, while the language adds typed devices, operational roles, capabilities, signals, constraints, and bindings to external facility systems.

STATUS OF PALS

The PALS project, which includes documentation and parsers, is hosted open-source on GitHub under the organization name `pals-project` [4]

Recent development has concentrated on stabilizing the core schema and documentation. PALS uses a language-neutral schema presented primarily through YAML examples, with `.pals.yaml` as the current reference file style. Recent updates define the top-level PALS root structure, clarify version and provenance fields. Lattice elements are organized around named element types with well defined parameter groups. Recent work involves development of the RF, particle distribution, Twiss, aperture and multipole parameter groups. Several design decisions have been clarified, including ordered lists versus dictionaries, element and parameter matching syntax, include file behavior, inheritance, constants, expressions, and the distinction between the base (User defined) and expanded lattice representations.

In parallel with the schema work, three reference implementations of the standard are currently under active development, in Python, C++, and Julia, all available as separate repositories under the same `pals-project` organization. Some of these, notably the Python implementation, are already being leveraged by beam tracking codes such as ImpactX, and broader adoption across simulation codes is planned and actively encouraged.

The PALS project follows an open-source governance model with a Steering Committee, currently consisting of David Sagan (Cornell) and Jean-Luc Vay (LBNL), that oversees the project along with a Technical Committee for day-to-day governance. Weekly discussion meetings, that are open to all interested parties, promotes community involvement. Anyone interested in PALS in any capacity is encouraged to participate.

PALS has attracted attention from a wide range of institutions as attested to by this paper’s author list. Projects include the UH Mark V FEL, the BNL Booster-to-AGS transfer line, the BNL EIC project, Jefferson Lab’s CEBAF machine, the LCLS and FACET-II injectors, and the SNS Linac and ring.

RELATION TO THE US DOE GENESIS MISSION

The U.S. Department of Energy (DOE) Genesis Mission provides an important organizing context for the expanded role of PALS. The Genesis Mission is a national effort to connect advanced computing, experimental facilities, AI systems, and scientific datasets into an integrated platform for accelerated discovery [5]. For accelerator facilities, this vision immediately raises an interoperability challenge. AI tools developed at one laboratory should not have to be rebuilt from first principles at each new machine simply because the local lattice files, control channels, databases, and operational conventions differ. The Genesis Mission therefore motivates the case for community standards that make accelerator information portable, queryable, and usable by both humans and AI systems.

Two Genesis-related accelerator efforts illustrate this need. The NARAD project is developing facility-agnostic, context-rich data infrastructure for nuclear physics accelerators in a form that can be generalized across Office of Science facilities. In this model, PALS provides the community accelerator language: ordered beamline elements, machine hierarchy, containment relationships, element classes, parameters, and extensible semantic relationships. NARAD uses this PALS-described structure as the anchor for AI-ready operational context, including device capabilities, semantic signal descriptions, constraints, archived-data associations, and facility-specific bindings to control-system channels. Thus, a quadrupole, cavity, BPM, or loss monitor is represented both as a lattice element and as an operational object with machine-queryable quantities and endpoints. NARAD therefore treats PALS as a portability layer above facility-specific databases and controls abstractions, not as a replacement for them.

This capability is especially important for MOAT, a multi-office, multi-facility effort to develop shared AI tools for accelerator design, operations, training, and scientific workflows. Its initial technical demonstrations have focused on Osprey, an agentic AI framework for safety-critical control-room environments [6]. Osprey demonstrates how language-model agents can be connected to facility tools under explicit

safety policies, human approval gates, and auditable execution paths. However, for such frameworks to travel across laboratories, they must resolve operator intent into the correct local machine objects, signals, and control endpoints.

One specific challenge is channel finding: mapping a natural-language request, such as a request for orbit correctors, BPM readbacks, cavity gradients, or radiation monitors, onto the specific control-system channels that implement those quantities. Modern accelerator facilities can expose tens to hundreds of thousands of channels, and the mapping from physical intent to channel name is often facility-specific, historical, and only partly documented [7]. A robust knowledge-base query layer is therefore not an optional convenience for AI-assisted operations; it is a prerequisite for safe and portable tool use.

PALS can serve as the common accelerator language and structural backbone for this ecosystem. NARAD uses and extends it to create AI-ready accelerator data products; MOAT depends on such shared representations so that agentic workflows can be reused across facilities; and Osprey provides one operational framework in which these representations can be exercised. In this sense, the evolution of PALS from a lattice standard to a language standard is not just a change in terminology, but rather reflects a practical requirement imposed by a national scale AI effort. Accelerator tools must operate on shared meaning, not only on shared files. This is the practical distinction between a lattice standard and a language standard: the former exchanges accelerator models, while the latter also supports reasoning about what machine objects mean and how they are used.

UH MARK V FEL USE CASE

The University of Hawai'i Mark V Free Electron Laser, currently being refurbished at UH Mānoa, uses PALS as the lattice interchange format for its multi-code simulation platform, FELsim. A single PALS-conformant lattice file drives three simulation backends: an in-house linear-matrix code for fast matching optimization, COSY INFINITY [8,9] for arbitrary-order differential-algebraic transfer maps with fringe fields, and RF-Track (CERN) for particle tracking with space charge. The COSY INFINITY backend is driven through `pals2cosy` [10], an open-source converter that consumes both official PALS and FELsim lattice formats. PALS was the natural target when FELsim transitioned from an Excel-based lattice description to a structured interchange format. Current work covers the 40 MeV electron transport line; the lattice is being extended to support end-to-end modeling of the full UH FEL, including the upstream linac and the FEL undulator.

NARAD-PALS USE CASES

The Brookhaven Booster-to-AGS transfer line provides an early NARAD-PALS demonstration. A PALS-conformant lattice is generated from the Bmad [11] optics model, while device capabilities, operational signals, and controls bindings are extracted from the legacy ADO control-system in-

frastructure and validated against NARAD semantics [12]. The resulting representation links physics lattice elements to machine-queryable control-system associations. It can also be converted back into Bmad for tracking and simulation studies, demonstrating a closed loop among Bmad, PALS, NARAD semantics, and facility controls metadata.

A complementary Jefferson Lab NARAD use case applies the same pattern to CEBAF operations. PALS describes the CEBAF beamline structure and accelerator objects, while NARAD links SRF cavities, BPMs, magnets, and radiation monitors to semantic signals such as cavity gradient setpoints and readbacks, orbit readbacks, magnet strengths, and loss or neutron-monitor data. This enables AI-assisted queries that assemble the relevant devices and archived signals for studies such as gradient management, radiation reduction, fault response, and operations optimization, while preserving CEBAF-specific EPICS and archiver bindings as local facility implementation details.

OUTLOOK AND CONCLUSIONS

PALS is moving from a lattice-exchange specification toward a broader community language for accelerator information. Near-term work will focus on maturing the schema and governance process, extending the standard to describe operational objects, device capabilities, semantic signals, controls bindings, and facility-specific constraints, and validating these ideas through concrete use cases as detailed above.

Important open questions remain in how best to represent specialized accelerator components, control-system abstractions, archived data relationships, and safety-relevant operational context in a way that is both precise and practical for facilities to adopt. Continued participation from accelerator physicists, controls experts, software developers, and AI/ML researchers will therefore be essential. As accelerator facilities become more complex and increasingly connected to AI-assisted workflows, PALS can provide the shared semantic foundation needed for portable simulation, analysis, optimization, and operations tools across the accelerator community.

ACKNOWLEDGMENTS

Work supported in part by U.S. Department of Energy (DOE) grant DE-SC0025351, and by DOE's Genesis Mission, DOE Office of Science, and Office of Advanced Scientific Computing Research's ModCon under awards AC02-05CH11231, DE-FG02-08ER41546, DE-SC0018636, DE-SCL0000127, and DE-SC0025583.

Support in part comes from the CAMPA collaboration, a project of the U.S. Department of Energy, Office of Science, Office of Advanced Scientific Computing Research and Office of High Energy Physics, Scientific Discovery through Advanced Computing (SciDAC) program.

REFERENCES

- [1] Multi-Office Accelerator Team Collaboration, “MOAT”, 2026, <https://moat-ai.github.io>,
- [2] C. Mitchell *et al.*, “A community effort toward a Particle Accelerator Lattice Standard (PALS)”, in *Proc. NAPAC2025*, Sacramento, CA, USA, Aug. 2025, pp. 350–353. [doi:10.18429/JACoW-NAPAC2025-TUP004](https://doi.org/10.18429/JACoW-NAPAC2025-TUP004)
- [3] C. Tennant, “A Core Ontology for Particle Accelerators: Interoperable Data and Workflows Across Facilities”, *arXiv*, 2025. [doi:10.48550/arXiv.2512.00868](https://doi.org/10.48550/arXiv.2512.00868)
- [4] Particle Accelerator Language Standard (PALS), <https://github.com/pals-project>
- [5] The Genesis Mission, “The Genesis Mission”, <https://genesis.energy.gov>, 2026,
- [6] T. Hellert, J. Montenegro, and A. Sulc, “Osprey: production-ready agentic AI for safety-critical control systems”, *APL Machine Learning*, vol. 4, p. 016103, 2026. [doi:10.1063/5.0306302](https://doi.org/10.1063/5.0306302)
- [7] T. Hellert *et al.*, “From Natural Language to Control Signals: A Conceptual Framework for Semantic Channel Finding in Complex Experimental Infrastructure”, *arXiv*, 2025. [doi:10.48550/arXiv.2512.18779](https://doi.org/10.48550/arXiv.2512.18779)
- [8] M. Berz and K. Makino, “COSY INFINITY and its use for singlepass and multipass systems”, *Microscopy*, p. dfag016, 2026. [doi:10.1093/jmicro/dfag016](https://doi.org/10.1093/jmicro/dfag016)
- [9] K. Makino and M. Berz, “COSY INFINITY version 9”, *Nucl. Instrum. Methods*, vol. 558, pp. 346–350, 2006. [doi:10.1016/j.nima.2005.11.109](https://doi.org/10.1016/j.nima.2005.11.109)
- [10] E. Valetov, “pals2cosy: PALS lattice to COSY INFINITY (COSYScript) converter”, 2026, <https://github.com/evvaletov/pals2cosy>,
- [11] D. Sagan, “Bmad: A Relativistic Charged Particle Simulation Library”, *Nucl. Instrum. Methods Phys. Res., Sect. A*, vol. 558, pp. 356–359, 2006. [doi:10.1016/j.nima.2005.11.001](https://doi.org/10.1016/j.nima.2005.11.001)
- [12] W. Lin *et al.*, “AI-Ready Lattice Representation and ML Optimization for the BNL Booster-to-AGS Transfer Line”, presented at IPAC'26, Deauville, France, May 2026, paper MOP6311, this conference.