



HYP_Link

MIT Announces New, Easier HPC Programming Language

Jaclyn Ludema and Bob Sorensen
April 2025

RECENT DEVELOPMENT

USA-based MIT's Computer Science and Artificial Intelligence Laboratory (CSAIL) recently [unveiled a new programming language](#) for high-performance computing (HPC) called Exo 2. This language allows HPC programmers to explicitly control how the compiler generates code by creating scheduling operations outside the compiler. This approach may foster faster user-directed code execution while facilitating the creation of reusable scheduling libraries. MIT claims this approach may boost flexibility and promote the reuse of scheduling code across different applications and hardware platforms.

ANALYST COMMENT

Making HPC/AI more approachable is a pervasive goal across industry, government and academic institutions. When asked to list their barriers to furthering their AI capabilities, 38.8% of HPC/AI sites listed their level of in-house AI expertise and 31.1% of sites listed the complexity of integrating AI models into existing workloads. These challenges highlight the need for tools that simplify optimization and deployment, enabling broader adoption of HPC/AI technologies.

Exo 2 is categorized as a user-schedulable language (USL), a group of programming languages that have had limited applicability due to their relatively fixed set of scheduling operations, making it difficult to reuse scheduling code across different code blocks. MIT researchers suggest that Exo 2 will be more adaptable, enabling programmers to write up to 100 times less scheduling code while still achieving performance on par with leading HPC libraries, including Basic Linear Algebra Subprograms (BLAS) that power many machine learning applications. Exo 2 introduces safe, composable primitives that enable automation, code reuse, and fine-tuned control, empowering engineers to better optimize workloads.

Program languages like Exo 2 could lower barriers to HPC/AI development by reducing the amount and complexity of scheduling code required, making it accessible to smaller teams or individual developers. Improving USL adaptability could support diverse use cases, enabling developers to reuse scheduling code and focus on innovation rather than repetitive tasks. By striking a balance between automation and user control, USLs like Exo 2 could enable teams to handle the complexities of novel hardware accelerators, such as tensor cores and TPUs, which typically require specialized and often complex optimizations. This innovation could accelerate progress in AI and HPC, enabling a wider audience to tackle complex challenges and unlock new possibilities.

Copyright Notice

Copyright 2025 Hyperion Research LLC. Reproduction is forbidden unless authorized. All rights reserved. Visit www.HyperionResearch.com to learn more. Please contact 612.812.5798 and/or email info@hyperionres.com for information on reprints, additional copies, web rights, or quoting permission.