



EnergyPlus-MCP: A model-context-protocol server for ai-driven building energy modeling

Han Li, Yujie Xu, Tianzhen Hong^{*}

Building Technology and Urban Systems Division, Lawrence Berkeley National Laboratory, Berkeley, CA 94720, USA

ARTICLE INFO

Keywords:

Model context protocol
EnergyPlus
MCP server
AI
Building simulation
LLM

ABSTRACT

Traditional building energy modeling with the EnergyPlus building performance simulation engine requires domain expertise, programming skills, and intensive manual efforts limiting its effective adoption. This paper introduces EnergyPlus-MCP, the first open-source Model Context Protocol (MCP) server specifically designed for EnergyPlus simulation workflows, establishing a new foundational infrastructure for AI-driven building energy modeling. The MCP server implements a layered architecture with 35 specialized tools spanning model management, editing and analysis, HVAC and other systems configuration inspection, and simulation execution, enabling Large Language Models to interact with EnergyPlus through conversational interfaces. The server addresses critical workflow barriers by automating model validation, streamlining energy efficiency measures modification, and providing intelligent output management with interactive visualization. Through practical demonstrations using a multi-zone building retrofit analysis, we show how the EnergyPlus-MCP server significantly reduces manual efforts while maintaining full simulation rigor. By providing accessible natural language interfaces to sophisticated building energy analysis, this approach enables scalable deployment of simulation expertise across public and private organizations, educational institutions, and research teams, fundamentally transforming traditional building energy modeling practices.

Code metadata

Current code version	v0.1.0
Permanent link to code repository	https://github.com/LBNL-ETA/EnergyPlus-MCP
Permanent link to reproducible Capsule	https://github.com/LBNL-ETA/EnergyPlus-MCP
Legal Code License	BSD
Code versioning system used	Git
Software code languages, tools, and services used	mcp eppy matplotlib networkx pandas plotly graphviz
Compilation requirements, operating environments & dependencies	Python \geq 3.10 Docker
Developer documentation/manual	libraries in pyproject.toml file API reference in README.md file
Support email for questions	hanli@lbl.gov

1. Introduction

1.1. Background

Buildings account for more than one-third of global energy consumption and significantly contribute to greenhouse gas emissions, highlighting the critical need for improving their efficiency and sustainability [1]. Building Energy Modeling (BEM), utilizing detailed physics-based simulations to assess building performance and evaluate improvement opportunities to reduce energy use and costs, is central to addressing this global challenge. Among various BEM tools, EnergyPlus, developed by the U.S. Department of Energy, has emerged as an open source prominent building simulation engine that accurately models thermal dynamics, HVAC system operations, and environmental interactions. Such simulations can realize energy savings of approximately 20–60 % compared to conventional building codes, directly contributing to global building sector targets [2].

Parallel to advancements in BEM, artificial intelligence (AI), specifically Large Language Models (LLMs), have recently experienced

^{*} Corresponding author.

E-mail address: thong@lbl.gov (T. Hong).

<https://doi.org/10.1016/j.softx.2025.102367>

Received 18 June 2025; Received in revised form 7 September 2025; Accepted 11 September 2025

Available online 17 September 2025

2352-7110/© 2025 The Author(s). Published by Elsevier B.V. This is an open access article under the CC BY-NC-ND license (<http://creativecommons.org/licenses/by-nc-nd/4.0/>).

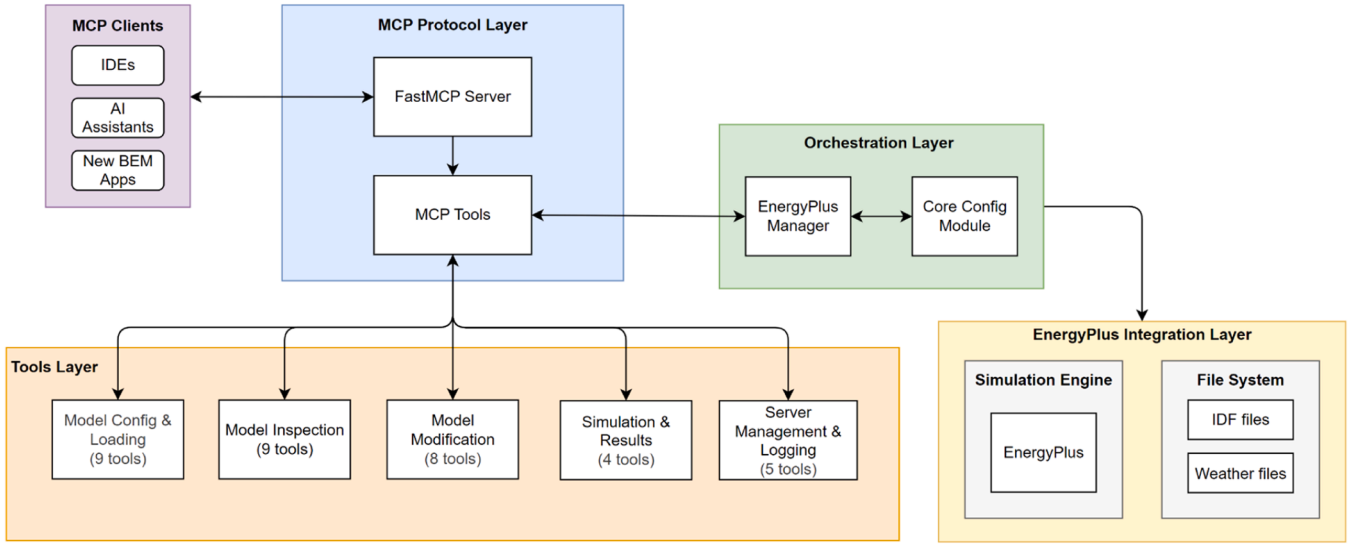


Fig. 1. EnergyPlus-MCP Software Architecture.

transformative development. These foundational models, trained on vast quantities of textual and structured corpora, possess powerful capabilities in interpreting, generating, and processing complex language and data patterns. Notably, research like the EPlus-LLM project has demonstrated the potential of these AI technologies to automate the creation of EnergyPlus input files directly from natural language descriptions, dramatically reducing manual effort [3]. Furthermore, the emergence of the Model Context Protocol (MCP) [4] provides a standardized interface enabling AI tools to dynamically and accurately interact with domain-specific software via structured, context-aware schemas, significantly enhancing reliability and interoperability [5].

1.2. Motivation

Despite its evident benefits, the traditional workflow of BEM, particularly when using sophisticated tools like EnergyPlus [6], involves significant challenges. Constructing accurate and detailed EnergyPlus input files (IDFs) requires domain expertise, programming skills, considerable manual effort, and meticulous management of a vast array of parameters, from building geometry and material properties to operational schedules and control strategies. Additionally, the iterative process of debugging and refining models can be cumbersome, presenting substantial barriers even for seasoned professionals. This complexity limits EnergyPlus adoption in large-sized or budget-rich building projects, and often results in a pronounced performance gap, where actual building energy performance deviates significantly from predicted values, consequently undermining stakeholder confidence and limiting the broader adoption of simulation-driven approaches [7].

Advanced utilities such as EMS [8], Python API [9], and FMU [10] enhance EnergyPlus and enable complex co-simulation and programmable controls of the simulation process. However, these tools have a steep learning curve and can sometimes be difficult to set up and debug. EMS allows for customized control logic to be incorporated in the simulation, but has limited flexibility due to the constraints of the native scripting language ERL, which lacks the flexibility and abundance of external libraries of modern programming languages like Python or Java. Thus, it can only implement simple control algorithms with no external interactions. Python API and FMU facilitate the programmable control of the simulation process and interaction with other simulation environments, but they could add substantial overhead and hinder the scalability of the co-simulation.

Emerging AI technologies, particularly LLMs, have shown promise in automating EnergyPlus workflows, with projects like EPlus-LLM

demonstrating natural language-driven input file generation [3]. However, direct LLM approaches face fundamental limitations, including inconsistent results, a lack of extensibility, a security gap, and an inability to support iterative workflows required by building professionals. The Model Context Protocol (MCP), introduced by Anthropic in November 2024, has emerged as a transformative solution by addressing the scalability challenge where each AI model traditionally required separate custom integrations for every software tool or data source [11]. MCP transforms this complexity into a simple architecture through its open, model-agnostic protocol, enabling standardized AI-software interactions [12]. This model-agnostic design allows the EnergyPlus-MCP server to work seamlessly with any MCP-compatible LLMs, without requiring separate integrations for each AI system. The protocol's rapid adoption, with over 5000 active servers and support from major AI providers within six months, demonstrates its technical merit in providing structured, secure integration where direct LLM approaches often fail [13]. Unlike monolithic LLM solutions, MCP facilitates robust interactions between AI agents and simulation engines through validated tools and standardized interfaces, ensuring both reliability and scientific rigor while dramatically reducing technical barriers for building energy modeling workflows.

1.3. Organization

This paper introduces EnergyPlus-MCP, a novel and first open-source Model Context Protocol server specifically designed for EnergyPlus, addressing critical challenges in current BEM practices. The remainder of the paper is organized as follows: Section 2 describes the software architecture and functionalities. Section 3 presents illustrative examples demonstrating practical applications in streamlining EnergyPlus workflows. Section 4 discusses potential impacts and benefits of building energy simulations. Section 5 provides concluding remarks and outlines future research directions.

2. Software description

2.1. Software architecture

As illustrated in Fig. 1, EnergyPlus-MCP implements a multi-layered architecture that cleanly separates client interfaces, protocol communication, service orchestration, domain logic, and simulation engine integration. This architectural design follows the principle of separation of concerns, enabling robust interactions between diverse MCP clients

Table 1
EnergyPlus-MCP Tool Categories and Functions.

Category	Tool Name	Primary Function
Server Management & Logging (5 tools)	get_server_status	Monitor server health and configuration
	get_server_configuration	Get detailed server configuration
	get_server_logs	Retrieve recent server log entries
	get_error_logs	Get recent error log entries
	clear_logs	Clear/rotate log files with backup
Model Config & Loading (9 tools)	copy_file	Intelligent file copying with path resolution
	load_idf_model	Load and validate EnergyPlus IDF files
	list_available_files	Browse sample files, example files, and weather data
	get_model_summary	Extract building, site, and simulation control information
	validate_idf	Comprehensive model validation with error reporting
	check_simulation_settings	Review simulation control and run period settings
	discover_hvac_loops	Identify all HVAC loops in the model
	get_loop_topology	Map the detailed loop topology and components
	visualize_loop_diagram	Generate interactive HVAC system diagrams
Model Inspection (9 tools)	list_zones	Enumerate thermal zones with spatial properties
	get_surfaces	Extract building envelope surface details
	get_materials	Analyze material properties and constructions
	inspect_schedules	Comprehensive schedule object analysis
	inspect_people	Detailed analysis of People objects with occupancy calculations
	inspect_lights	Comprehensive analysis of Lights objects with power calculations
	inspect_electric_equipment	Analysis of ElectricEquipment objects with power calculations
	get_output_variables	List defined output variables and meters
	get_output_meters	Discover available energy meters with units and frequencies
	modify_people	Modify People objects with flexible targeting
	modify_lights	Modify the Lights objects with calculation methods
	modify_electric_equipment	Modify ElectricEquipment objects with calculation methods
	modify_simulation_control	Update simulation control parameters
Model Modification (8 tools)	modify_run_period	Adjust simulation time periods
	change_infiltration_by_mult	Scale infiltration rates by a multiplier
	add_window_film_outside	Apply window films to exterior surfaces
	add_coating_outside	Add exterior coatings to walls or roofs
	add_output_variables	Add output variables with intelligent validation
Simulation & Results (4 tools)	add_output_meters	Add output meters with flexible specification formats
	run_energyplus_simulation	Execute complete EnergyPlus simulations
	create_interactive_plot	Generate a web-based visualization from outputs

and EnergyPlus while maintaining clear boundaries between different functional responsibilities. The layered approach facilitates maintainability, testability, and extensibility by isolating protocol-specific concerns from domain expertise and simulation engine dependencies.

The architecture comprises five distinct layers serving complementary roles in the system. The MCP Protocol Layer handles all standardized communication with MCP clients, including IDEs, AI assistants, and emerging BEM applications, managing message routing, tool discovery, and transport abstraction through the FastMCP server framework. The Orchestration Layer transforms protocol-level requests into domain-specific operations via the EnergyPlus Manager, providing input validation, error handling, and response formatting while coordinating with the Core Config Module for system-wide configuration management. The Tools Layer encapsulates all building energy modeling expertise across five functional domains, maintaining complete independence from protocol concerns while orchestrating complex workflows involving model validation, component analysis, HVAC system evaluation, and simulation management. The EnergyPlus Integration Layer manages all interactions with the simulation engine and file system operations, handling IDF and weather file processing, library integrations, and simulation execution in a controlled manner.

This architectural foundation emphasizes modularity and extensibility through clear interface definitions and dependency management. The design enables new building energy modeling capabilities to be added at the tools level without requiring changes to protocol handling or integration layers. Configuration management flows consistently through all layers via a hierarchical system, while comprehensive error handling and logging facilitate debugging and monitoring across the entire system. The containerized deployment strategy ensures consistent environments and simplifies scaling, making the architecture suitable for both development workflows and production deployments across diverse MCP client ecosystems.

2.2. Core functionalities

EnergyPlus-MCP provides five primary functional domains that collectively enable comprehensive building energy modeling workflows through AI-driven interfaces. The implementation utilizes the FastMCP framework to register an initial set of 35 specialized tools, with a central manager class orchestrating all domain-specific operations through the eppy library for direct EnergyPlus integration. Table 1 summarizes the available tools organized by functional category, demonstrating comprehensive coverage of building energy simulation workflows.

These tools implement consistent patterns for error handling, logging, and response formatting, ensuring reliable programmatic access across diverse AI agent implementations. The functional categories address the complete building energy modeling workflow from initial model loading and validation through simulation execution and results visualization. Server management tools provide enterprise-grade operational capabilities, including health monitoring, debugging support, and log management essential for production deployments. Model configuration and loading tools enable robust file operations with intelligent path resolution and comprehensive HVAC system discovery. Model inspection tools extract detailed information across all building components, including advanced occupancy analysis, equipment assessment, and schedule parsing with value extraction capabilities. Model modification tools support systematic changes to building parameters, envelope properties, and simulation settings with flexible targeting options. Simulation and results tools orchestrate complete EnergyPlus execution workflows with intelligent output configuration, validation, and interactive visualization capabilities.

2.3. Implementation highlights


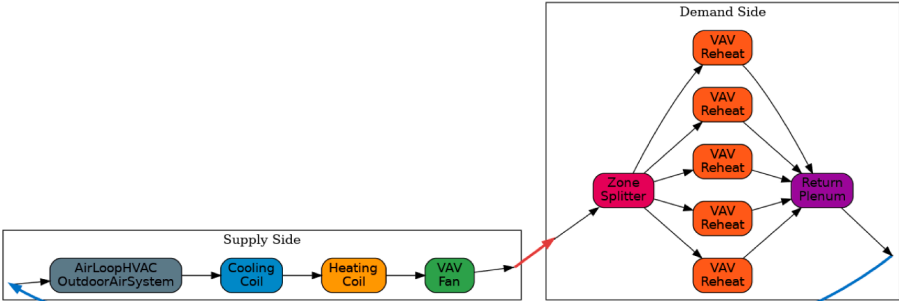
EnergyPlus-MCP is implemented in Python version 3.12 with core dependencies including FastMCP [14] for JSON-RPC-based MCP

Table 2
Queries and actions from the scenario setup.

Human Queries	Agent Actions
Q1. Can you list the available files from the EnergyPlus example files directory? I'm specifically looking for models with '5 Zone' in their names. Also, check which default weather files we have.	The agent invoked the `list_available_files` tool, which found 94 IDF files whose names contain "5Zone" from EnergyPlus's ExampleFiles folder, and found 5 EPW files.
Q2. Copy the "5ZoneAirCooled.idf" model and the Chicago weather data to the `examples` directory.	The agent invoked the `copy_file` tool twice to copy IDF and EPW files to the target directory, respectively.

protocol communication and automatic tool discovery, and eppy [15] for object-oriented EnergyPlus IDF file manipulation and simulation execution. The server implements structured logging, comprehensive error handling with domain-specific exception types, and hierarchical configuration management supporting environment variables and deployment-specific settings.

Table 3
Queries and actions from model inspections.

Human Queries	Agent Actions
Q3. Load the 5ZoneAirCooled.idf model and show me the basic model information, including building details, site location, and simulation settings.	<p>The agent invoked the <code>load_idf_model</code> and <code>get_model_summary</code> tools, which provide comprehensive building metadata, revealing a 5-zone commercial building with specific construction details, climate settings, and simulation parameters.</p>
	<div><div><div><div><div></div><div>GitHub Copilot</div></div><div><div>> Used 2 references</div><div><div>> Ran <code>get_model_summary</code> – energyplus (MCP Server)</div></div></div><div>Here is the basic model information for <code>5ZoneAirCooled.idf</code>:</div></div><div><div>Building Details:<ul style="list-style-type: none">Name: BuildingNorth Axis: 30°Terrain: CityLoads Convergence Tolerance: 0.04Temperature Convergence Tolerance: 0.4Solar Distribution: FullExteriorMax Warmup Days: 25Min Warmup Days: 6</div><div>Site Location:<ul style="list-style-type: none">Name: CHICAGO_IL_USA TMY2-94846Latitude: 41.78Longitude: -87.75Time Zone: -6.0Elevation: 190.0 m</div><div>Simulation Settings:<ul style="list-style-type: none">Do Zone Sizing Calculation: YesDo System Sizing Calculation: YesDo Plant Sizing Calculation: YesRun Simulation for Sizing Periods: NoRun Simulation for Weather File Run Periods: YesDo HVAC Sizing Simulation for Sizing Periods: NoMax Number of HVAC Sizing Simulation Passes: 1</div></div><div>EnergyPlus Version: 25.1</div></div></div>
Q4. Discover all HVAC loops in this model and try to visualize the topology of the air-side system. Plot it as a PNG.	<p>The agent invoked the <code>discover_hvac_loops</code> tool to discover all HVAC loops in the model. After identifying a "VAV Sys 1" loop that matched our requirement for "air-side loop", it invoked the <code>get_loop_topology</code> tool to get the topology. Finally, it invoked the <code>visualize_loop_diagram</code> tool to generate the diagram below.</p> <div></div>

The core functionality is organized into modular components supported by specialized utility modules to achieve different functionalities: occupancy analysis with thermal comfort calculations, lighting system evaluation, equipment assessment, schedule parsing with value extraction, and output variable management. HVAC system analysis employs NetworkX [16] for graph-based topology algorithms and Graphviz [17] for generating hierarchical diagrams with component-level detail. Data processing utilizes pandas [18] for simulation output handling, while Plotly [19] generates interactive HTML visualizations with time-series analysis capabilities.

The deployment strategy centers on Docker [20] containerization with pre-configured EnergyPlus version 25.1.0 installation, ensuring easily reproducible and consistent environments across different operating systems. The development container includes Node.js 20 [21] to support MCP development tools and web-based interfaces. This design approach provides an extensible foundation through an initial set of building blocks that can accommodate additional simulation engines or analysis capabilities without requiring modifications to the core MCP

Queries and actions from energy efficiency modifications.

(continued on next page)

Table 4 (continued)

Q7. With the modified model, apply window film to all exterior windows with U-value of 3.5 W/m ² ·K and SHGC of	The agent first calculated the new power levels, then invoked the 'add_window_film_outside' and 'add_coating_outside' tools, respectively, to apply changes to the model. It then saved the modified model as directed.
0.35 to reduce solar heat gain. Add reflective coating to all exterior walls with a solar absorptance of 0.3 to minimize heat absorption. Save the final modified model with all energy efficiency measures as '5ZoneAirCooled_improv.ed.idf'	<div>I'll apply the window film and reflective coating to the modified model to create the final improved version with all energy efficiency measures.</div> <div>> Ran add_window_film_outside – energyplus (MCP Server) ✓</div> <div>> Ran add_coating_outside – energyplus (MCP Server) ✓</div> <div>Perfect! I've successfully applied all the energy efficiency measures to create the comprehensive improved model. Here's a summary of what was accomplished:</div> <div>Final Improved Model: 5ZoneAirCooled_improved.idf</div> <div>Energy Efficiency Measures Applied:</div> <div>1. Internal Load Reductions (from previous modification):</div> <div><ul style="list-style-type: none">Lighting Power: Reduced by 30% (7,500W → 5,250W)Electric Equipment: Reduced to 80% for all zones except SPACE3-1 (5,000W → 4,211.2W)</div> <div>2. Window Film Application:</div> <div><ul style="list-style-type: none">Applied to: All 4 exterior windows (WF-1, WR-1, WB-1, WL-1)U-value: 3.5 W/m²·KSHGC: 0.35 (significant reduction in solar heat gain)Visible Transmittance: 0.66 (maintains good daylighting)Construction: All windows now use cons_outside_window_film_QhnbForAkI</div> <div>3. Reflective Wall Coating:</div> <div><ul style="list-style-type: none">Applied to: All exterior wall surfaces using WD01 layerSolar Absorptance: 0.3 (reduced from 0.78 - 61% reduction)Thermal Absorptance: 0.9 (unchanged for proper heat transfer)</div>

interface implementation. The modular architecture establishes important groundwork for future development and integration with emerging building energy modeling workflows.

3. Illustrative examples

In this section, we demonstrate EnergyPlus-MCP through a simple energy retrofit analysis workflow using the 5ZoneAirCooled.idf model from EnergyPlus example files. The demonstration uses VSCode’s dev container environment running the MCP server, with GitHub Copilot serving as the MCP client to facilitate seamless natural language interactions between the LLM agent and comprehensive building energy simulation tools. In each subsection, we briefly describe the goals, human queries, and the agent’s actions.

3.1. Scenario setup

Our demonstration begins with an energy retrofit analysis of a multi-zone building. The model is a one-story building with five conditioned zones and one return plenum. The conditioned zones include four perimeter zones and one interior zone. We start by exploring available example models and selecting appropriate files for our analysis. Our queries and the agent’s actions are summarized in Table 2 below.

3.2. Model inspection and analysis

With our baseline model prepared, we begin the analysis by understanding the building’s current configuration and systems. Our goal is to extract key building characteristics, including basic model information, HVAC system topology, and generate visual representations to better

understand the system architecture before implementing energy-saving measures (Table 3).

3.3. Energy efficiency measures

Having established an understanding of the building’s characteristics and systems, we now implement targeted energy efficiency measures. In this demonstration, we consider reducing internal loads through lighting efficiency improvements and envelope modifications to minimize solar heat gain, demonstrating how the MCP tools enable rapid evaluation and implementation of energy-saving measures with built-in validation and error checking (Table 4).

3.4. Simulation and results

To compare the energy consumption between the baseline and modified models, we conduct comparative simulations of both models. Our analysis focuses on summer conditions when cooling loads are highest, and our envelope modifications are anticipated to have maximum impact. We systematically configure output variables to track lighting energy consumption, equipment loads, and solar heat gains, then execute simulations and generate interactive visualizations to assess retrofit effectiveness (Table 5).

4. Impact

As illustrated in Fig. 2, EnergyPlus-MCP has the potential to support a comprehensive ecosystem that bridges sophisticated building energy simulation capabilities with AI-enabled natural language interfaces. The architecture supports diverse stakeholders through specialized

Table 5

Queries and actions from the simulation and results.

Human Queries	Agent Actions																																																												
Q8. Modify the run period for both models to simulate summer months only (June through August).	The agent invoked the <code>`modify_run_period`</code> twice to change the run period for both models to June 1 - August 31.																																																												
Q9. Discover all available output variables and identify those related to lighting energy, electric equipment consumption, and solar heat gains through windows and surfaces.	The agent invoked the <code>`get_output_variables`</code> tool to discover all possible output variables in the model. Behind the scenes, this tool ran a short simulation for just one day, checked the output variable file (*.rdd), and then found 624 potential variables. It then scanned through them and suggested a set of variables for our analysis, shown below:																																																												
	<div><div>Recommended Output Variables for Energy Efficiency Analysis</div><div>For comparing the original vs. improved models, I recommend adding these key variables:</div><div><div>Output:Variable,*,Lights Electricity Energy,hourly;</div><div>Output:Variable,*,Electric Equipment Electricity Energy,hourly;</div><div>Output:Variable,*,Surface Window Transmitted Solar Radiation Energy,hourly;</div><div>Output:Variable,*,Zone Windows Total Heat Gain Energy,hourly;</div><div>Output:Variable,*,Surface Outside Face Solar Radiation Heat Gain Energy,hourly;</div></div></div>																																																												
Q10. Add these relevant variables to both models, then run simulations with the Chicago weather file.	The agent invoked the <code>`add_output_variables`</code> tool to add the suggested variables, and then invoked <code>`run_energyplus_simulation`</code> to run simulations with both models.																																																												
Q11. Read the simulation outputs from both scenarios, compare their end-use consumptions, and calculate the savings.	<div>The agent read the output files and summarized the comparison between the baseline and improved scenarios, shown below:</div> <div><div>Energy Consumption & Heat Gain Comparison: Baseline vs Improved Model</div><div><div><div><div><div><div></div><div>Energy End Uses Comparison (3-Month Summer Period: June-August)</div></div></div><table><tr><th>Category</th><th>Baseline Model (GJ)</th><th>Improved Model (GJ)</th><th>Difference (GJ)</th><th>% Change</th></tr><tr><td>Interior Lighting</td><td>20.53</td><td>14.37</td><td>-6.16</td><td>-30.0% </td></tr><tr><td>Interior Equipment</td><td>12.04</td><td>10.14</td><td>-1.90</td><td>-15.8% </td></tr><tr><td>Cooling</td><td>10.17</td><td>7.60</td><td>-2.57</td><td>-25.3% </td></tr><tr><td>Heating</td><td>2.71</td><td>2.85</td><td>+0.14</td><td>+5.2% </td></tr><tr><td>Fans</td><td>1.50</td><td>1.04</td><td>-0.46</td><td>-30.7% </td></tr><tr><td>Pumps</td><td>1.29</td><td>1.02</td><td>-0.27</td><td>-20.9% </td></tr></table></div><div><div><div><div></div><div>Sensible Heat Gain Comparison (Total Facility)</div></div><table><tr><th>Heat Gain Source</th><th>Baseline Model (GJ)</th><th>Improved Model (GJ)</th><th>Difference (GJ)</th><th>% Change</th></tr><tr><td>Lights Sensible Heat Addition</td><td>20.534</td><td>14.373</td><td>-6.161</td><td>-30.0% </td></tr><tr><td>Equipment Sensible Heat Addition</td><td>12.036</td><td>10.137</td><td>-1.899</td><td>-15.8% </td></tr><tr><td>Window Heat Addition</td><td>28.534</td><td>16.578</td><td>-11.956</td><td>-41.9% </td></tr><tr><td>People Sensible Heat Addition</td><td>8.243</td><td>8.260</td><td>+0.017</td><td>+0.2% </td></tr></table></div></div></div></div></div>	Category	Baseline Model (GJ)	Improved Model (GJ)	Difference (GJ)	% Change	Interior Lighting	20.53	14.37	-6.16	-30.0%	Interior Equipment	12.04	10.14	-1.90	-15.8%	Cooling	10.17	7.60	-2.57	-25.3%	Heating	2.71	2.85	+0.14	+5.2%	Fans	1.50	1.04	-0.46	-30.7%	Pumps	1.29	1.02	-0.27	-20.9%	Heat Gain Source	Baseline Model (GJ)	Improved Model (GJ)	Difference (GJ)	% Change	Lights Sensible Heat Addition	20.534	14.373	-6.161	-30.0%	Equipment Sensible Heat Addition	12.036	10.137	-1.899	-15.8%	Window Heat Addition	28.534	16.578	-11.956	-41.9%	People Sensible Heat Addition	8.243	8.260	+0.017	+0.2%
Category	Baseline Model (GJ)	Improved Model (GJ)	Difference (GJ)	% Change																																																									
Interior Lighting	20.53	14.37	-6.16	-30.0%																																																									
Interior Equipment	12.04	10.14	-1.90	-15.8%																																																									
Cooling	10.17	7.60	-2.57	-25.3%																																																									
Heating	2.71	2.85	+0.14	+5.2%																																																									
Fans	1.50	1.04	-0.46	-30.7%																																																									
Pumps	1.29	1.02	-0.27	-20.9%																																																									
Heat Gain Source	Baseline Model (GJ)	Improved Model (GJ)	Difference (GJ)	% Change																																																									
Lights Sensible Heat Addition	20.534	14.373	-6.161	-30.0%																																																									
Equipment Sensible Heat Addition	12.036	10.137	-1.899	-15.8%																																																									
Window Heat Addition	28.534	16.578	-11.956	-41.9%																																																									
People Sensible Heat Addition	8.243	8.260	+0.017	+0.2%																																																									

applications: Architecture, Engineering, and Construction (AEC) professionals can perform rapid design iterations and energy performance comparisons, building owners gain access to retrofit analysis tools, policy makers can automate code compliance verification for standards like ASHRAE 90.1 [22] and Title 24 [23], while researchers and educators benefit from streamlined model lifecycle management and automated documentation capabilities. The MCP server layer provides standardized tool access to building data, simulation management, and post-processing capabilities, enabling seamless integration with existing AI assistants and development workflows.

The broader impact extends beyond technical implementation to fundamental workflow transformation in the building industry. By providing conversational interfaces to EnergyPlus while maintaining full simulation rigor, EnergyPlus-MCP has the potential to enable more scalable deployment of building energy expertise across organizations managing large building portfolios, educational institutions, and research teams conducting extensive parametric studies. This approach has the potential to reduce traditional technical barriers without

compromising scientific accuracy, making sophisticated building energy analysis accessible to a broader range of professionals and accelerating the adoption of evidence-based design and operation practices in the built environment.

5. Limitations

While EnergyPlus-MCP provides comprehensive building energy modeling capabilities, several limitations should be acknowledged. The current tool set focuses on model analysis and modification rather than complete model creation from scratch, which remains a complex challenge requiring domain expertise. The quality of AI-driven interactions heavily rely on the capabilities of the MCP client and the user's ability to formulate appropriate queries. Additionally, while our tools automate many workflows, users still need sufficient BEM knowledge to validate results. The protocol communication layer introduces minor computational overhead compared to direct EnergyPlus usage, and the server inherits any limitations present in the underlying EnergyPlus simulation

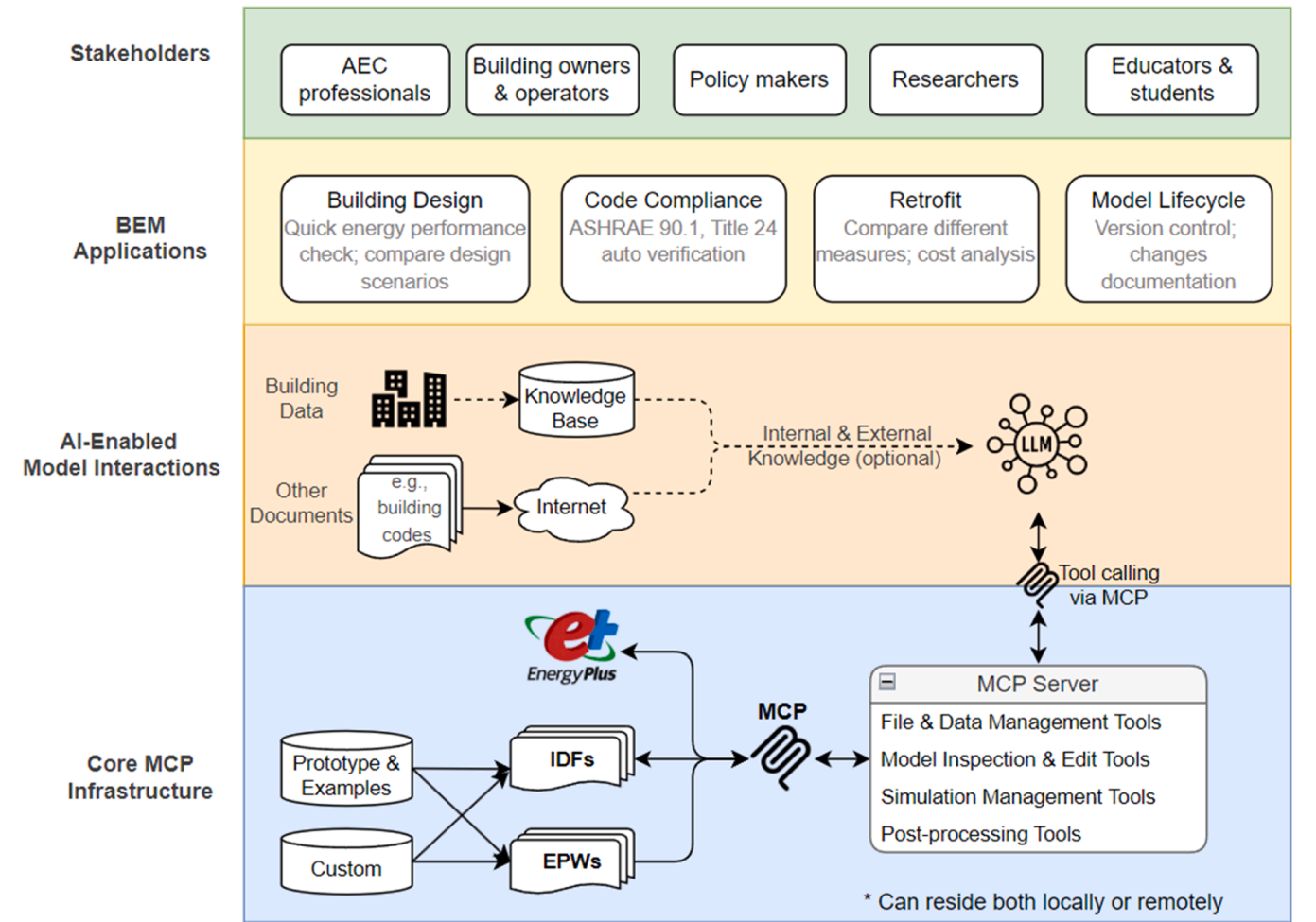


Fig. 2. EnergyPlus-MCP's role in the BEM ecosystem.

engine.

6. Conclusion and future work

This paper introduced EnergyPlus-MCP, the first novel and open-source Model Context Protocol server for EnergyPlus building energy simulation. Through an initial set of 35 tools spanning file management, model analysis, HVAC system intelligence, and simulation execution, EnergyPlus-MCP addresses key barriers in traditional building energy modeling. The layered architecture enables robust interactions between AI assistants and simulation capabilities while maintaining a clear separation between protocol communication and domain expertise. Our demonstrations with an example model showed how the server streamlines workflows from model validation to energy efficiency modifications and simulation management, reducing manual effort without compromising scientific rigor. By democratizing EnergyPlus access through conversational interfaces, EnergyPlus-MCP enables scalable deployment of building energy expertise across organizations, educational institutions, and research teams, making sophisticated analysis accessible to broader audiences.

Future developments of the EnergyPlus-MCP server include expanding the toolset with more sophisticated features such as geometry creation and editing, HVAC loop creation and editing, and advanced schedule modification capabilities. The server architecture enables integration into diverse building energy applications, including model calibration, retrofit analysis, parametric simulations, model predictive control, and real-time digital twin implementations. These applications can leverage the standardized MCP interface to automate complex

workflows and enable systematic building performance optimization. Looking ahead, EnergyPlus-MCP provides a foundation for developing next-generation building energy modeling software with built-in AI assistants as MCP clients, fundamentally transforming how building professionals interact with simulation tools and enabling more intuitive, conversational approaches to building energy analysis and design.

Acronyms

AI	Artificial intelligence
AEC	Architecture, Engineering, and Construction
API	Application programming interface
BEM	Building energy modeling
FMU	Functional mockup unit
HVAC	Heating, ventilation, air-conditioning
IDE	Integrated development environment
IDF	EnergyPlus input data file
LLM	Large language model
MCP	Model context protocol

Code availability

The EnergyPlus-MCP repo is publicly available on GitHub at: <https://github.com/LBNL-ETA/EnergyPlus-MCP>

CRediT authorship contribution statement

Han Li: Writing – review & editing, Writing – original draft,

Visualization, Software, Methodology, Formal analysis, Conceptualization. **Yujie Xu:** Writing – review & editing, Writing – original draft, Software, Methodology. **Tianzhen Hong:** Writing – review & editing, Supervision, Methodology, Funding acquisition, Conceptualization.

Declaration of competing interest

The authors declare there are no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

This work was supported by the Assistant Secretary for Energy Efficiency and Renewable Energy, Office of Building Technologies of the United States Department of Energy, under Contract No. DE-AC02-05CH11231. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the Department of Energy.

References

- [1] Jean-Pascal T. Buildings are the foundation of our energy-efficient future. World Econ Forum; 2021. <https://www.weforum.org/stories/2021/02/why-the-buildings-of-the-future-are-key-to-an-efficient-energy-ecosystem/>. accessed June 10, 2025.
- [2] EnergyPlus essentials n.d. <https://energyplus.readthedocs.io/en/stable/essentials/essentials.htm> (accessed August 7, 2025).
- [3] Jiang G, Ma Z, Zhang L, Chen J. EPlus-LLM: a large language model-based computing platform for automated building energy modeling. Appl Energy 2024; 367:123431. <https://doi.org/10.1016/j.apenergy.2024.123431>.
- [4] Get started with the Model context protocol (MCP). Model context protoc n.d. <https://modelcontextprotocol.io/introduction> (accessed June 10, 2025).
- [5] Edwards B. MCP: the new “USB-C for AI” that’s bringing fierce rivals together. Ars Tech; 2025. <https://arstechnica.com/information-technology/2025/04/mcp-the-new-usb-c-for-ai-thats-bringing-fierce-rivals-together/>. accessed June 10, 2025.
- [6] DOE. EnergyPlus: creating a new-generation building energy simulation program. Energy and Buildings, 33(4): 319-331, 2001.
- [7] de Wilde P. The gap between predicted and measured energy performance of buildings: a framework for investigation. Autom Constr 2014;41:40–9. <https://doi.org/10.1016/j.autcon.2014.02.009>.
- [8] EMS Application Guide - EnergyPlus 25.1 documentation n.d. <https://energyplus.readthedocs.io/en/stable/ems-application-guide/ems-application-guide.html> (accessed June 10, 2025).
- [9] EnergyPlus Python API - EnergyPlus 25.1 documentation n.d. <https://energyplus.readthedocs.io/en/latest/api.html> (accessed June 10, 2025).
- [10] Coupling EnergyPlus with functional mock-up units for co-simulation: external Interface(s) Application Guide - EnergyPlus 25.1 n.d. https://energyplus.net/assets/nrel_custom/pdfs/pdfs_v25.1.0/ExternalInterfacesApplicationGuide.pdf (accessed June 10, 2025).
- [11] Hou X., Zhao Y., Wang S., Wang H. Model context protocol (MCP): landscape, security threats, and future research directions 2025. <https://doi.org/10.48550/arXiv.2503.23278>.
- [12] Ray P.P. A survey on Model Context Protocol: architecture, State-of-the-art, challenges and future directions 2025.
- [13] Narajala V.S., Habler I. Enterprise-grade security for the Model context protocol (MCP): frameworks and mitigation strategies 2025. <https://doi.org/10.48550/arXiv.2504.08623>.
- [14] Lowin J. jlowin/fastmcp 2025.
- [15] santoshphilip. santoshphilip/epmy 2025. (accessed June 10, 2025).
- [16] Hagberg A., Schult D., Swart P. networkx/networkx: network analysis in Python, 2025 (accessed June 1, 2025).
- [17] Lück C., Frings S., Reuben. Graphviz/graphviz 2025.
- [18] McKinney W. others. Pandas: a foundational Python library for data analysis and statistics. Python High Perform Sci Comput 2011;14:1–9.
- [19] Kruchten N., Seier A., Parmer C. An interactive, open-source, and browser-based graphing library for Python 2024. <https://doi.org/10.5281/zenodo.14503524>.
- [20] Docker I. Docker 2020.
- [21] nodejs/node 2025.
- [22] American Society of Heating, Refrigerating and Air-Conditioning Engineers (ASHRAE) and Illuminating Engineering Society (IES). ANSI/ASHRAE/IES Standard 90.1-2022: energy Standard for buildings except low-rise residential buildings 2022.
- [23] California Building Standards Commission. California Building Standards Code, Title 24 2022.