Agentic Compute–Energy Co-Optimization System Using Beckn Protocol

Idea Submission (Problem 2)

1 Team Information

Team Name: Watt's Up Agents
Institution / Organization: Imperial College London
Team Members (2–4): Jensen Chen, Yuhao Yue, Yi Zhou, Letian Jiang
Contact Emails: jc4822@ic.ac.uk, ay923@ic.ac.uk, mz2023@ic.ac.uk, letian.jiang@seh.ox.ac.uk
Discord Usernames: jc4822, daddy_01943

2 Problem Focus

We address Problem 2, Compute–Energy Convergence in a DEG World, by making AI workloads a controllable, verifiable source of demand flexibility for the grid.

3 Solution Overview

We propose a lightweight agentic co-optimisation layer that schedules AI workloads when and where electricity is cheapest and cleanest without breaking workload constraints. Compute clusters expose flexible job slots as Beckn catalog items, each tagged with universal asset and location identities plus a machine-readable energy profile and flexibility class. Grid agents publish an information plane of regional carbon intensity, electricity price, renewable share, congestion state, and flexibility incentives. An orchestration agent chooses the best compliant slot to minimise £ per inference under a carbon cap, then confirms execution through a Beckn search–select–confirm lifecycle. Every decision is written to tamper-evident audit logs so savings, emissions, and trade-offs are transparent and reusable across regions.

4 Technical Architecture

The MVP runs three cooperating agents behind a FastAPI service. The compute agent maintains a small catalogue of tasks with duration, deferral window, priority, allowed regions, and identifiers that make each task a first-class digital asset. It generates candidate execution windows and publishes them through `/search` as Beckn catalog entries containing timing, region `location_id`, estimated `power_kW`, and flexibility type. The grid agent simulates a per-hour, per-region information plane that includes carbon, price, renewables share, congestion flags, and a simple flexibility reward, and validates a slot via `/confirm`. The orchestration agent enumerates feasible slots, queries the information plane, computes a weighted score combining carbon, cost, and reward, selects the best slot that respects the carbon cap, and submits it for confirmation. Optional LLM advice may explain decisions, but the final scheduler is rule-based for interpretability and auditability.

5 Agent Workflow

A scheduling request triggers `/search` to retrieve feasible job slots from the compute agent. The orchestration agent evaluates these against the grid information plane, reserves a preferred slot with `/select`, and sends the resulting order to the grid agent through `/confirm`. The grid agent confirms if the slot meets carbon, price, and congestion rules and aligns with renewable availability or flexibility incentives; otherwise it rejects with a reason. If rejected, the orchestration agent retries within the deferral window. Each step emits a hash-chained audit entry containing the task identity, chosen window, grid state, objective values, and justification.

6 Business Model & Impact

Customers include data-centre and cloud operators who want lower operating costs and carbon-aware service guarantees, and grid operators or aggregators seeking dependable flexible demand. The system can be offered as a SaaS control plane or embedded into schedulers, priced per managed GPU-hour or per verified flexibility event. Operators gain reduced £ per inference by avoiding peak-price windows and earn additional revenue when workloads shift into grid-preferred periods. Grids benefit from smoother demand, reduced congestion, and lower renewable curtailment. Because identities, descriptors, and audit trails are standardised and verifiable, the approach scales across vendors and geographies with minimal integration effort, matching the DEG goal of open, modular digital infrastructure.

7 References / Inspiration

This submission is inspired by the Digital Energy Grid vision of universal identity, machine-readable metadata, and verifiable transactions, and by Beckn Protocol's open workflows for interoperable agent coordination.

8 Declarations

IP & Licensing: Submitted under MIT Commons License.
Submission Format: 1–2 page PDF uploaded via Dora Hacks.
Deadline: 23/11/25 17:00 GMT.