Digital Energy Grid Hackathon — Idea Submission Template

1. Team Information

Information	Details
Team Name	COOP: Computing Optimisation
Institution / Organization	University of Oxford
Team Members (2–4)	Yuzhong Luo, Founder and CEO Xuzhou Ren, CFO, VP Business/Product Jerry Jin, CTO, VP Engineering Yashasvika Ahuja, Marketing Manager
Contact Emails	yuzhong.luo@trinity.ox.ac.uk tiger.ren@trinity.ox.ac.uk jerry.jin@lmh.ox.ac.uk yashasvika.ahuja@balliol.ox.ac.uk
Discord Usernames	pig8 _redbananaz jerrydjin

2. Problem Focus

Select one problem statement your solution addresses:

Problem 2: Compute–Energy Convergence in a DEG World

3. Solution Overview (max 150 words)

Briefly describe your idea. What problem are you solving, and how does your agentic solution create value in the Digital Energy Grid ecosystem?

We're building a multi-agent orchestration layer that treats AI compute as a flexible energy asset in the Digital Energy Grid. Today, data centres schedule jobs without real-time awareness of grid carbon intensity, local constraints, or cheaper regions, wasting flexibility and money. Our solution uses a hierarchy of agents (global, regional, local) to continuously rebalance AI workloads across geographies based on £/inference, capacity, and carbon caps. Local compute agents both communicate their job slots and costs via Beckn catalogs and use real-time energy and carbon signals, micro-managing all operational decisions. The global grid agent then assigns, shifts, or defers tasks to low-carbon, low-cost regions, logging all trade-offs for audit. This creates value by cutting compute costs, reducing emissions, and giving DSOs/grid operators predictable, controllable demand-side flexibility from AI workloads.

4. Technical Architecture (max 200 words or diagram)

Explain the AI agent you plan to build and how they will solve the problems and create value, while leveraging Beckn and DEG components.

Include:

- Key agents and their roles
- Data sources / APIs / models used (e.g., Beckn Sandbox, AI models)
- Orchestration or coordination logic

Include all assumptions and references for your architecture design.

(Optional: include a simple diagram or link to a visual model in your submission)

[A simple diagram or link to a visual model illustrating the technical architecture]

We implement three AI agents:

- **Global Grid Agent** – Runs a periodic optimisation (e.g. hourly) over all queued tasks and regional summaries. Objective: minimise £/inference under a carbon-intensity cap and capacity constraints. It issues task assignments/reassignments to regional agents, and defers jobs into future "buckets" when carbon or network constraints are tight.
- **Regional Agents (~40)** – One per macro-region (e.g. NA West, EU, MENA, OCE). They aggregate Beckn catalogs from local/compute agents, maintain top-N lowest cost/inference options, and forecast regional demand. They report costs, capacities, and current workloads to the global agent.

- **Local Compute Agents** – City or data-centre cluster level. They monitor energy data (prices, carbon intensity, renewables, storage state of charge), detect abnormal events, and forecast near-term energy conditions and compute availability.

Data sources / APIs: Beckn Sandbox for publishing and discovering compute slots; synthetic/standard APIs for grid prices, carbon intensity, and data-centre telemetry; simple time-series models for energy and workload forecasts.

Assumptions: All compute nodes and DSOs expose Beckn-compliant APIs; DEG provides a messaging bus for events. A web dashboard visualises agent decisions and logs.

(Diagram: Global agent ↔ Regional agents ↔ Local/compute agents ↔ Grid/energy APIs + Beckn network.)

5. Agent Workflow (max 150 words)

Describe how your agent works along with relevant Beckn protocol flows, where applicable in your context.

Local compute agents publish their available job slots, prices, capacities, and carbon characteristics as Beckn **/catalog** (search/on_search). When a new batch of tasks arrives (via Beckn order/search from an upstream client), the **global grid agent** queries regional catalogs (search/on_search), compares offers, and uses select/on_select + confirm/on_confirm to place workloads with specific compute providers.

Regional agents translate global assignments into specific local jobs, using Beckn **/status** and **/on_status** to track execution and **/update /on_update** for rescheduling within their region.

If energy or carbon conditions change (e.g. local spike), local agents notify regional agents, which propagate updated catalogs. On the next optimisation cycle, the global agent may issue Beckn **/cancel** or **/update** requests to defer or re-route tasks to other regions. All decisions, inputs, and trade-offs are logged for later audit and visualised in the dashboard.

6. Business Model & Impact (max 150 words)

Outline the potential business model, value capture, and stakeholders who benefit (utilities, data centers, aggregators, consumers, etc.).

Include any sustainability, scalability, or revenue aspects.

We operate a subscription-based B2B platform. Compute providers and data centre operators pay a monthly fee for access to our orchestration network. Additionally, a per-task usage fee is charged, dynamically priced based on the task’s computational intensity, time sensitivity (deadline proximity), local grid carbon intensity, and regional energy congestion. These same factors are used by our agents to optimise placement, deferral, and migration decisions. Value is created through lower £/inference, carbon-aware scheduling, and access to flexibility markets (e.g. P415). Stakeholders like DSOs gain visibility into a large, schedulable load class and can trigger demand-side response. Aggregators benefit from integrating this new flexibility asset class. Our architecture is scalable by design—new nodes and energy markets can be added without re-architecting the system. Sustainability is embedded: workloads shift to cleaner energy periods, improving grid efficiency and lowering emissions without new infrastructure.

7. References / Inspiration (optional)

Mention any prior work, open datasets, or publications you’re building upon.

8. Declarations

- IP & Licensing: Submitted under MIT Commons License
- Submission Format: 1-2 page PDF uploaded via Dora Hacks
- Deadline: Date Calendar event