

## **1) Business / Organization**

### **Focus**

- Deliver all JFP roadmap items on schedule in FY25.
- Define how JFP can be delivered as an independent module (aligned to business/marketing).

#### **a) Timely & Aligned Delivery of JFP**

- Kept regular sync-ups with product/platform to refine priorities, remove blockers, and verify release readiness per milestone.
- Created the complete JFP service package (containerized, Podman-ready).
- Delivered JFP Monitoring CM commands.
- Set up JFP infrastructure (RPMs, configs) and partnered with Monitoring for data collection.
- Brought up slurmrestd (HTTPS + UNIX socket) from initial build scripts and RPMs.
- JFP deployment for HPCM 1.15: service package, CM commands, RPMs, and end-to-end infra.
- Slurm REST API fully enabled (HTTPS + UNIX socket) to support JFP delivery.
- Install scripts & step-by-step guide for Slurm REST to ensure repeatable setup.

#### **b) Extensive Dataset Exploration & Integration**

- Evaluated public HPC datasets (Grid Workloads Archive, Google Cluster traces) to strengthen robustness.
- Collaborated with HPCD/platform teams to obtain anonymized HPE test-cluster job metadata for real-world characteristics.
- Built standardized preprocessing to handle CSV / JSON / Kafka; consistent feature extraction across datasets.
- Labs dataset intake initiated (with Jeff; pushed Sergey) to unblock data dependencies.

#### **c) Results Derivation & Cross-Dataset Comparisons**

- Implemented a repeatable evaluation pipeline across datasets.
- Tracked validation metrics (precision, recall, F1) per dataset to test generalizability.
- Shared EDA reports with stakeholders; reviewed with Sergey each cycle.

#### **d) Seamless Integration of XTIME & MLOps for JFP**

- Reusable hyperparameter-tuning module (random/grid) via XTIME for XGBoost.

- Logged model versions, metrics, and artifacts to MLflow for reproducibility and visibility.
- Enabled modular save/load of .ubj models so JFP auto-loads the latest model.
- JFP ↔ XTIME integration with public datasets; code improvements to accelerate delivery.

### **Challenge/Learning**

- Coordinating multiple teams and data sources required a tight schedule and clear acceptance criteria.

## **2) Customer**

### **Focus**

- Define simplified JFP service offerings aligned to customer needs.
- Collaborate with NREL to validate and extend customer value.

#### **a) Self-Contained JFP Module**

- Packaged JFP as a standalone container with clear, stable interfaces so customers can plug it into existing stacks without rework.

#### **b) End-to-End Predictive Pipeline (Customer-oriented)**

- Standardized metadata schemas across datasets.
- Preprocessing to clean, normalize, and encode job parameters.
- Per-dataset XGBoost training and validation with integrated online inference.
- Outputs feed predictions to dashboards/schedulers for real-time decisions.
- Guidance to reduce misconfigurations (e.g., excessive memory, wrong partition).

#### **c) Customer Discovery & Partnerships**

- Held stakeholder discussions to align use cases and scope across teams.
- Authored HPCM Copilot use-case documents; ran intake with L2 Support, CPE, USS, and Support teams.
- Partnered with CPE on Cray MPI/libfabric tunables on Slingshot.
- Worked with Dean on LNM / isolation scenarios.
- Ongoing collaboration with Sun Chun on CPE use cases.

#### **d) NREL Collaboration (my role)**

- Proactively drove the engagement: pushed internal owners to connect with NREL, prepared context, and opened a technical thread with Hilary Egan and Kevin Menear.
- Sourced and trialed NREL datasets for JFP; iterated with feedback until we had a viable path for failure/resource/memory predictions.

## **Challenge/Learning**

- Customer needs were diverse—packaging a small, adoptable offer required saying no and sequencing asks.

## **3) Innovation**

### **Focus**

- Drive new technical capability across the stack (scheduler, data, ML, agents) and publish results.
- Create reusable building blocks that raise the team's innovation velocity.

#### **a) Complex System Integration (under HPCM constraints)**

- Integrated SLURM manager with slurmdbd and slurmrestd within HPCM limits; resolved auth (MUNGE/JWT) and inter-service wiring using reproducible runbooks.
- Codified troubleshooting into step-wise playbooks so others can replicate and scale the setup quickly.

#### **b) API Exploration & Feature Extraction**

- Built POC scripts against Slurm OpenAPI and SlurmDB REST to fetch live and historical job metrics.
- Validated parity/freshness vs. CLI (sacct) to ensure completeness and real-time accuracy of features.

#### **c) Unified Feature Layer & Submit-time Analytics**

- Customized internal modules to unify job metadata from three sources: sacct, slurm/v0.0.42/job/{id} (OpenAPI), and slurmdb/v0.0.42/jobs (DB-REST).
- Enabled a single feature layer for both real-time and batch predictions, including submit-time failure analytics.

#### **d) AI/Agentic Prototypes & Community Contributions**

- Prototyped a Copilot-style MCP server + chat path to interact with REST endpoints; informed the JFP Agent design.
- Designed the JFP Agent with clear roles and interfaces; explored on-box LLM deployment options for inference.
- Submitted papers to SC25 and TechCon2026 to share approach and results with the community.
- Outlined data-ops innovations: Predictive Disk Usage Alerts (VictoriaMetrics) and Kafka message-rate anomaly detection.

## **Challenge/Learning**

- Proving new patterns (REST parity, unified feature layer, agent POCs) needed extra validation and crisp documentation to win adoption.

## **4) Leadership Capability Development**

### **Focus**

- Foster a collaborative, knowledge-sharing culture that accelerates delivery and innovation.
- Grow team capabilities in AI/ML and modern data/agent stacks through hands-on enablement.

#### **a) Knowledge Sharing & Onboarding**

- Authored and socialized implementation guides and troubleshooting notes for the SLURM API ecosystem (slurmrestd setup, auth, packaging).
- Ran working sessions with Monitoring and QE to onboard teams and improve replication speed.

#### **b) Technical Mentoring & Practice Growth**

- Mentored peers on XGBoost tuning, MLflow pipelines, and integrating real-time inference flows.
- Applied and shared feature-engineering techniques to improve predictive accuracy and reduce false negatives.

#### **c) Cross-Team Alignment & Program Leadership**

- Initiated and drove the HPCM Copilot proposal; aligned stakeholders and converted it into a milestone-tracked program—establishing the foundation for JFP as an independent module.
- Facilitated stakeholder discussions to align use cases and scope across L2 Support, CPE, USS, and Support teams; captured decisions in concise docs.

#### **d) Upskilling & Community**

- Completed one week of HPCM training to deepen system understanding.
- Strengthened skills in MCP, Generative AI, and AI agents; shared practical takeaways with the team.

### **Challenge/Learning**

- Docs alone weren't enough—hands-on sessions and follow-ups were key to lasting adoption across teams.