

Project Status: Agentic AI for Engineering Simulation

Date: December 18, 2025

Focus: Infrastructure Hardening & Data Retention

Progress (Current State)

Architecture: Transitioned to a Distributed Monorepo (Dockerized).

Components Deployed:

- Orchestrator: Streamlit frontend + OpenAI (currently 1-shot mapping).
- MCP_Server: FastAPI state manager.
- Reporter_Agent: Automated result capturing.
- Database: PostgreSQL with shared Pydantic schemas.

Status: The system works logically, but the execution engine (Abaqus) is not yet fully integrated into the containerized flow.

The Plan (This Week: Week 1)

Goal: Solve the "Headless" execution and ensure raw data is saved (Dr. Li's Requirement).

A. Abaqus Worker Deployment (The Blocker)

- Strategy Selection: Decide between Docker + xvfb (harder, cleaner) vs. Native VM (easier, messier).
- Implementation:
- Configure Fea_worker to launch Abaqus without a GUI.
- Verify connection to the FlexNet license server from inside the worker environment.
- Smoke Test: Run a simple beam deflection .inp file through the worker to verify the pipe.

B. Data Retention (The "Surrogate Model" Asset)

- Storage Infrastructure: Provision an Azure Blob Storage container (or S3 bucket).
- Worker Update: Modify Fea_worker Python script to:
 - Locate the .odb (output database) and .dat files after a run.
 - Upload them to Blob Storage.
 - Return the Storage URL to the MCP Server instead of just the text report.
- DB Schema: Add artifact_url column to the Jobs table.

C. Service Integration

- Network Check: Ensure the Orchestrator can trigger the Worker and receive the "Job Started" signal immediately.

Notes & Context

Professor's Feedback:

Crucial: Do not discard raw simulation files (.odb). We need them to train future proxy models.

Action: Storage costs are secondary; data preservation is primary.

Current Bottlenecks:

- Headless Abaqus: Running commercial FEA inside Docker is the main technical hurdle right now.
- Orchestrator Logic: currently simple; will need LangGraph soon, but after the pipeline is stable.