



南京大學
NANJING UNIVERSITY

2026 MicroTCA Workshop

Design of MicroTCA-based Digital Twin System for Accelerator Systems

A Layered Hybrid Architecture Integrating Real-Time Control, HPC/GPU Compute and AI Agents

MicroTCA / MTCA.4

EPICS

Digital Twin

AI & AI Agents

Hardware-in-the-Loop

Presenter: Paul Chu

September 15, 2026

01

Background & Motivation

Why large-scale accelerator facilities need a digital twin

02

Digital Twin Fundamentals & Objectives

Five-dimension model · five core objectives · scope

03

Requirements Analysis

Functional needs and key performance indicators

04

System Architecture

Five-layer technical architecture

05

MicroTCA Hardware Platform Feasibility

Latency · channel capacity · external compute invocation

06

Recommended Hybrid Architecture

MTCA.4 front-end + data platform + HPC/GPU/LLM back-end

07

AI & AI-Agent Integration

Prediction · tuning · LLM · agents from assistant to hands-free

08

Implementation, Cost & Risks

Two-phase roadmap · 2–3-person team · ~4.6M CNY

09

Conclusions & Outlook

Key takeaways and future directions

Background & Motivation

Why a digital twin for accelerator facilities?

Large-scale accelerator facilities

- Synchrotron light sources — SSRF, HEPS
- Spallation neutron sources — CSNS
- Heavy-ion accelerators — HIRFL, CIADS
- Free-electron lasers — DCLS, SXFEL
- Core national S&T infrastructure; decade-scale lifecycles

Shared challenges

- High complexity: many coupled subsystems (injector, ring, transfer lines, stations)
- Stringent operation: thousands of running hours / year, high stability and efficiency
- Heavy O&M burden: equipment ageing, complex spares, knowledge loss as experts retire

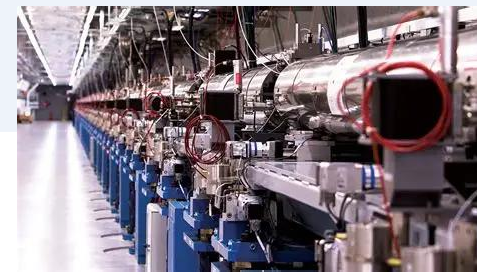
The limits of “manual monitoring + post-hoc troubleshooting”

- Cannot meet demands of intelligent operation, rapid optimization and knowledge retention
- Slow, reactive and heavily dependent on scarce senior expertise

Digital twin → a high-fidelity virtual replica of the physical facility

- Real-time mapping between physical and virtual worlds
- Simulation, prediction and what-if analysis
- Intelligent, data-driven decision support

MicroTCA (MTCA.4) provides the real-time data bridge between the facility floor and the twin backend.



Digital Twin Fundamentals & Objectives

Five-dimension model · value proposition

PE

Physical Entity

accelerator systems & devices

VM

Virtual Model

geometry · physics · behaviour
· data-driven models

Ss

Services

monitoring · simulation ·
tuning · diagnosis · training

Dd

Data

real-time · historical ·
simulation · configuration ·
knowledge

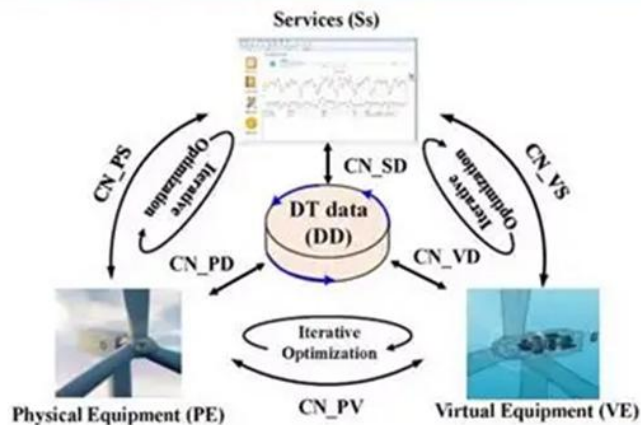
CN

Connections

EPICS CA/PVA · data bus · REST
API

Key technical elements

- Multi-scale modeling & simulation (macro → micro)
- Real-time data synchronization (sensors, EPICS, IoT)
- AI fusion: machine learning, deep learning, knowledge
- Visual interaction: 3D rendering, AR/VR/MR



Five Core Objectives & System Scope

What the twin must deliver



南京大學
NANJING UNIVERSITY

1

Real-time mapping

1:1 digital mirror of the facility; live beam parameters, equipment status, environment

2

Simulation & prediction

Multi-physics modeling (beam dynamics, EM, thermal, vacuum) + what-if scenarios

3

Intelligent optimization

AI-driven parameter optimization and intelligent beam tuning, incl. AI-Agent evolution

4

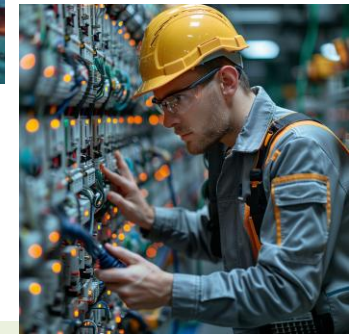
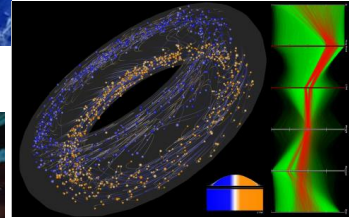
Fault diagnosis

Intelligent fault localization, root-cause analysis and remediation recommendations

5

Collaborative R&D

Virtual prototyping, **hardware-in-the-loop (HIL)**, training and knowledge management



System coverage

- Physical: injector (e-gun / ion source, RFQ, DTL) · main accelerator (linac / ring / cyclotron) · beam transfer lines · experimental / target stations · auxiliary systems (vacuum, magnet PS, RF, cooling, safety interlock)
- Digital: full 3D geometry (1:1) · physics simulation models · 1:1 mirror of all EPICS control variables (PVs) · historical data warehouse · knowledge base (docs, fault cases, expert experience)

Functional requirements

- **Data acquisition:** EPICS CA/PVA, thousands of concurrent PVs; OPC UA / Modbus / MQTT / REST
- **Physics simulation:** OPAL, MAD-X, Elegant, Geant4; multi-physics coupling
- **AI/ML:** LSTM/Transformer prediction, anomaly detection, Bayesian / RL optimisation
- **LLM:** RAG-based operations Q&A, auto report generation, fault recommendations
- **Visualization:** 1:1 3D model, beam trajectory, heatmaps; large screen / Web / mobile / AR-VR
- **HIL & training:** what-if analysis, virtual-IOC testing, operator training

Non-functional

- **Security:** 3-level network isolation, TLS, RBAC, full audit
- **Compatibility:** existing EPICS, Linux/Windows, standard interfaces
- **Scalability:** plugin modules, dynamic device access, horizontal scaling
- **Maintainability:** hot-swap hardware, containers, logging & monitoring

Key performance indicator	Requirement
Beam-signal real-time latency	< 10 ms end-to-end
Control command response	< 1 ms (real-time / interlock)
Data availability	≥ 99.99%
Data throughput	> 1,000,000 msg/s (Kafka)
Sampling / precision	MHz level · 16-bit+ ADC
Channel scale	hundreds of channels; thousands of PVs
Time synchronisation	PTP μs level or better (IEEE 1588)

Reliability: carrier-grade 99.999% at the real-time acquisition layer

Overall Architecture — Five-Layer Technical Stack

Mapping of the five-dimension twin framework onto the facility



南京大學
NANJING UNIVERSITY

Application Services

Run monitoring · simulation analysis · intelligent tuning · fault diagnosis · PdM · training · knowledge management — large screen / desktop / Web / mobile / AR-VR



Model & Algorithm Layer

L0 geometry · L1 physics · L2 behavior · L3 data-driven · L4 rule models — model registry, validation & orchestration



Data Platform Layer

InfluxDB / TimescaleDB · PostgreSQL · MinIO/S3 · Neo4j — Kafka pipeline + Flink/Spark streaming, ETL & data governance



Network & Communication

EPICS control network (CA/PVA) · 10GbE data network (Kafka / ZeroMQ / DDS) · compute network (InfiniBand / NVMe-oF) · NTP / PTP



Perception Layer

BPM · ICT/FCT · emittance monitors · screens · SEM grids · vacuum gauges · temperature / field / current sensors · radiation monitors · PLCs · RF

Hardware Platform Challenge

No single platform can satisfy all requirements



Seven platform families assessed: MicroTCA (MTCA.4) · PXIe · x86 server clusters · GPU clusters · FPGA PCIe · edge AI (Jetson) · cloud

Trade-offs between latency, compute power, AI capability, rendering and cost — a single platform cannot win on all

Platform	Best at	Limits
MicroTCA / MTCA.4	μ s latency · EPICS-native · HIL · hot-swap · 99.999%	No local physics / AI / LLM → callable externally
PXIe	Instrument DAQ, many channels	No GPU / AI; similar niche to MicroTCA
x86 server cluster	Data platform, applications, cost-effective	No hard real-time
GPU cluster	AI training/inference, physics acceleration, 3D render	Cost; not for control loops
FPGA PCIe	ns-level deterministic processing in a server	Ecosystem smaller than MTCA.4
Edge AI (Jetson)	Low-power on-site inference	No precise clock; light real-time
Cloud	Elastic scaling, on-demand GPU, batch simulation	Latency jitter; data-security concerns

► **Conclusion: a layered heterogeneous hybrid architecture is essential — MicroTCA is not an “island” full-stack platform but the high-speed data bridge between the facility floor and the twin backend.**

MicroTCA / MTCA.4 Technology Overview

PICMG standard · telco-grade · proven in accelerators

The standard

- MicroTCA (PICMG MTCA.0): open modular architecture, up to 12 AMC slots per chassis
- MTCA.4: physics extension with enhanced clock & trigger for accelerators
- Core modules: AMC (CPU / FPGA / ADC), MCH (management + clock + switch), μ RTM, power & cooling
- Backplane: PCIe Gen3/4 · SRIO · 10GbE; μ s-level inter-module latency
- PTP time sync 100 ns – 1 μ s; deterministic multi-chassis triggers

Why it fits accelerators

- EPICS-native: IOC runs on AMC (Linux / RTEMS / vxWorks)
- Proven at DESY, European XFEL, ESS — 10+ years of production use
- Carrier-grade reliability 99.999%; full hot-swap maintenance
- FPGA hard real-time (ns-level) + virtual-IOC HIL capability

External communication — the bridge to the twin

- AMC runs Linux → standard TCP/IP networking
- REST API · gRPC · Kafka client · EPICS CA/PVA
- SSH + Slurm → HPC job submission
- ZeroMQ / Socket.io for lightweight real-time links

→ *Standard interfaces let MicroTCA invoke external computing resources, compensating for its limited local computing power.*



Feasibility I — Real-Time Latency & Channel Capacity

MicroTCA meets the hard real-time requirements



Measured / typical link latency

- AMC backplane PCIe DMA: 1–5 μs
- AMC backplane SRIO: 1–3 μs ; MCH switching: 5–10 μs
- ADC $\rightarrow$ FPGA processing: 0.1–1 μs
- AMC $\rightarrow$ server over 10GbE: 50–200 μs
- End-to-end: 100–500 μs ; PTP sync: 100 ns–1 μs

Requirement satisfaction

- Beam-signal latency < 10 ms — fully met
- Control response < 1 ms — backplane level
- HIL closed loop < 10 μs — FPGA direct
- Data-bus uplink < 100 ms — satisfied
- External calls: MicroTCA data emission < 200 μs ; end-to-end pipeline must be evaluated

Channel capacity

- 6–12 AMC slots per chassis; 4–16 analog channels per AMC
- $\rightarrow$ 48–192 analog channels per chassis
- 16–64 digital I/O per AMC; thousands of EPICS PVs over the network
- 10+ chassis cascading $\rightarrow$ 10K–100K+ PV capacity

Satisfies a medium accelerator ($\sim 5,000$ PVs); most PVs are read from existing IOCs via Kafka rather than acquired on-board.

MicroTCA is the best fit for:

- High-real-time signals (BPM / ICT)
- Precision-synchronized channels
- HIL simulation interfaces

Feasibility II — External Computing Invocation

Turning compute limits into an elastic architecture (v2 revised)

Physics simulation

★★★★☆ (was ★)

Invoke external HPC via REST/gRPC:
MAD-X, OPAL, Elegant, Geant4;
MicroTCA packages & transmits data;
results return to the data platform

AI inference

★★★★☆ (was ★)

Kafka → GPU server; small models run
on AMC FPGA (< 1 ms); complex models
on back-end GPU cluster (ONNX /
TensorRT)

LLM support

★★★★☆ (was ☆)

Commercial LLM APIs with RAG
(embedding + vector DB); MicroTCA
triggers queries and displays results

Invokable external resources

- Local GPU workstation — small-model inference
- HPC cluster — physics simulation codes
- GPU inference server — AI / embeddings
- Cloud GPU instances — elastic training
- Commercial LLM APIs — Q&A, reports
- K8s microservices — twin applications



Interface methods & latency tiers

- REST / gRPC / Kafka / CA-PVA / ZeroMQ / SSH+Slurm
- < 1 ms FPGA local (lightweight models)
- < 100 ms same-room GPU inference
- < 10 s cloud API calls

Elastic: front-end hardware unchanged, back-end compute can be upgraded at any time — pay-as-you-go reduces initial investment.

Recommended Hybrid Architecture

MTCA.4 front-end + data platform + HPC/GPU/LLM back-end



南京大學
NANJING UNIVERSITY

Application Services

x86 / GPU (K8s)

3D visualization (Unity3D / UE5) · AR/VR · Web / mobile · API gateway · LLM Q&A · knowledge management



Models & Algorithms (external)

HPC + GPU + LLM API

MAD-X / OPAL / Geant4 physics · AI training & inference (PyTorch / TF) · commercial LLM API (RAG)



Data Middle Platform

x86 cluster (K8s)

Kafka > 1M msg/s · Flink / Spark · InfluxDB + PostgreSQL + MinIO · governance, ETL, metadata



Real-Time Control / Acquisition

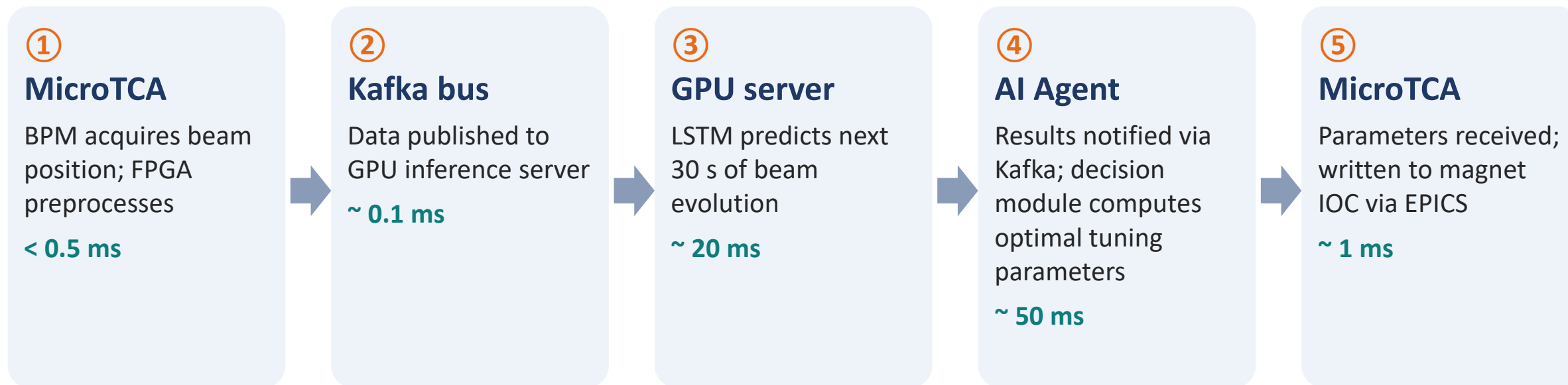
MTCA.4 + FPGA PCIe

EPICS IOC (CA/PVA) · BPM / ICT synchronized DAQ · HIL · FPGA lightweight inference

All layers coordinate via the Kafka data bus + REST API — data flows upward, control commands flow downward (with permission control).

End-to-End Data Flow: AI-Assisted Beam Tuning

Total closed-loop latency < 100 ms



Total end-to-end latency < 100 ms — a closed loop from beam measurement to magnet setpoint, closing on the real accelerator via EPICS.

Design notes:

- The same loop runs in simulation mode first (virtual IOC + virtual accelerator) to validate the policy before it touches real hardware.
- EPICS PV writes by the digital twin require explicit permission control (RBAC) and are fully audited.

AI Capabilities in the Digital Twin

Prediction · detection · optimization · language

Prediction

LSTM / Transformer multi-step forecasting of beam parameters (30–60 min horizon) · input: historical PV sequences

Anomaly detection

Autoencoder + threshold detection · operating-mode clustering · statistical drift & variance monitoring

Optimization

Bayesian optimization with GP surrogate · reinforcement-learning tuning · multi-objective (intensity, emittance, energy spread)

Transfer learning

Train in simulation → fine-tune on the real machine (sim-to-real) — fewer experiments on beam time

Large language models (LLM)

- RAG architecture: structured extraction of manuals, fault cases and SOPs → operations Q&A and natural-language data queries (“What is the beam current now?” → direct PV lookup)
- Automatic run-report generation · fault-handling recommendations · code assistance for data analysis scripts

AI Agents: From Assistant to Hands-Free Operation

The evolution ladder of agentic operation



Stage 1 — AI-assisted

Agent proposes; operator decides. Tuning recommendations, anomaly alerts, RAG answers.

Stage 2 — Semi-autonomous

Agent executes routine tuning under operator supervision; human approval for critical actions.

Stage 3 — Hands-free

Agent runs the machine autonomously within a defined safety envelope; full audit trail.

Agent operating loop (a copilot resident in the twin)

Perceive

PV streams · alarms · logs · beam state

Reason

LLM planner + physics-informed models + knowledge graph

Act

EPICS writes (RBAC) · HPC jobs · tool calls

Learn

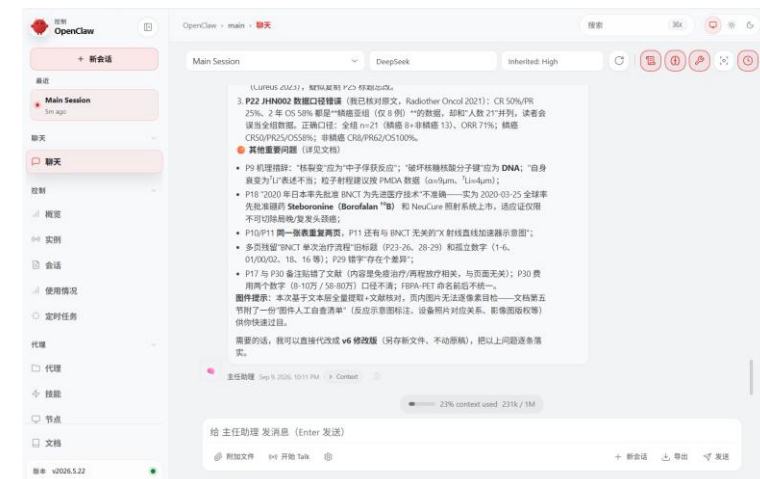
episodic memory · feedback · case library

Drivers of agent behavior

- SOP-driven: learn operation sequences from standard operating procedures
- History-driven: reuse successful tuning cases and past operator decisions
- Model-predictive: plan actions with physics models and predicted beam response; multi-objective optimization

Multi-agent view: Beam-Tuning Agent · Fault-Diagnosis Agent · Predictive-Maintenance Agent · Operations Copilot — orchestrated, with human-in-the-loop gates.

MicroTCA-based Digital Twin System for Accelerator Systems



Application scenarios

- **Intelligent tuning**: set beam target → agent plans, executes and replays multi-step tuning sessions; learns from successful cases
- **Fault diagnosis**: detect → locate (knowledge-graph reasoning) → root-cause analysis → LLM-generated handling advice; new cases auto-ingested
- **Predictive maintenance**: health score 0–100, remaining-useful-life (RUL) estimation, maintenance recommendations
- **Operations copilot**: “Why did the beam current drop?” → natural-language data query + root-cause summary + auto report
- **Multi-facility collaboration** (future): federated learning — data stays on site, models are shared

Safety guardrails

- **Twin-first**: every agent action is first validated in the digital twin / virtual IOC before touching real equipment
- **Write-access control**: EPICS PV writes require explicit permission (RBAC, least privilege)
- **Human-in-the-loop**: approval gates for critical or irreversible actions
- **Confidence thresholds**: low-confidence suggestions are flagged for human review, never auto-executed
- **Full audit trail of every decision and action**; simulation mode is isolated from production
- **LLM guardrails**: RAG with domain restriction + source citation; human review of critical outputs

The Digital Twin as AI Training & Test Bed

Sim-to-real · HIL · what-if — zero-risk experimentation



南京大學
NANJING UNIVERSITY

Safe reinforcement-learning environment

- The virtual accelerator is a full RL environment for tuning policies
- Train in simulation with domain randomisation → fine-tune on the real machine
- Cuts experiments that would otherwise consume scarce beam time

What-if scenario analysis

- “If this magnet current changes, what happens to the beam?”
- “If this vacuum pump fails, how does pressure evolve?”
- Heat-load safety checks in planned operating modes

Zero risk — no real accelerator time consumed.

Hardware-in-the-loop (HIL)

- Real controllers + virtual loads: magnet power supplies, RF cavities, vacuum systems, interlock PLCs
- Virtual IOCs substitute real devices for integration testing
- Validate new control algorithms and new devices before installation
- MTCA.4 + FPGA is the ideal HIL platform (ns-level determinism)

Continuous learning: new fault cases and successful tuning sessions flow back into the knowledge base; AutoML keeps models current.

Implementation Roadmap — Two Phases, 16–18 Mo.

Prototype first, full-featured product second · 2–3-person team



南京大學
NANJING UNIVERSITY

Phase 1 — Basic Prototype (6 months · ~1.66M CNY)

- First MTCA.4 chassis + EPICS IOCs; acquisition → Kafka → data platform; ≥ 1,000 PVs online
- 1:1 3D modelling of key equipment; large-screen + Web monitoring v1
- Milestones: design review (M1) → data chain live (M3) → viz v1 (M5) → acceptance (M6)
- Performance targets: latency < 10 ms, availability ≥ 99.99%

Phase 2 — Full-Featured Product (10–12 months · ~2.95M CNY)

- Physics simulation via external HPC; AI prediction & anomaly detection; LLM RAG Q&A
- Intelligent tuning assistant v1 (AI Agent); HIL platform
- Fault diagnosis & predictive maintenance; AR/VR (optional)
- Integration, security hardening, on-site deployment & acceptance (M16)

Overall schedule (Gantt view)



Strategy: the original three-phase roadmap (foundation / capability / intelligence) is folded into two phases — Phase 1 = foundation; Phase 2 = capability + intelligence.

Assurance: milestone reviews every 4–5 nodes · parallel work packages · schedule buffers (0.5 + 1 month) · front-load MicroTCA procurement, data-access agreements and HPC machine-time applications.

Team, Cost & Risks

A small team can deliver — if risks are managed



南京大學
NANJING UNIVERSITY

Team (standard option, 3 FTE)

- Project manager & system architect (1 FTE)
- Software & AI algorithm engineer (1 FTE)
- Hardware & embedded engineer (1 FTE)
- Accelerator physics consultant (0.5, Phase 2)
- 3D modelling outsourced as needed

2-person option → schedule extends ~30–40%; AR/VR & PdM become optional.

Top risks & mitigation

MicroTCA supply & long lead times	Early procurement, backup suppliers, x86 + FPGA PCIe fallback
Scarce MTCA.4 + EPICS expertise	Train core staff; collaborate with facility controls group; external consultants
Facility data access & quality	Early data agreements; develop with virtual IOC / simulated data
AI performance on real beam data	Simulation pretraining + transfer learning; phased rollout with gates
LLM hallucination	RAG + domain restriction + citation; human review of critical outputs
Budget overrun (GPU / cloud)	Three-tier budget control; pay-as-you-go cloud, monthly accounting

Cost item (standard tier, 10k CNY)	Phase 1	Phase 2	Total
Labour (51 person-months)	69.0	128.5	197.5
Hardware & equipment	71.0	107.0	178.0
Software / cloud / LLM API	3.0	18.0	21.0
Travel / testing / machine time	5.0	10.0	15.0
Management & contingency (~12%)	17.8	31.6	49.4
Total	165.8	295.1	460.9

Economy tier ~330–360 · Enhanced tier ~600–700 (10k CNY). Estimates are reference only.

Conclusions & Outlook

Key takeaways · future directions · thank you



Key takeaways

- MTCA.4 is the real-time data bridge: μ s latency, EPICS-native, HIL-ready, carrier-grade 99.999%
- Hybrid architecture wins: MTCA.4 front-end + x86 data platform + HPC/GPU/LLM back-end, coordinated via Kafka + REST
- External compute invocation turns MicroTCA's compute limits into an elastic, upgradeable architecture
- AI & agents: from monitoring $\rightarrow$ prediction $\rightarrow$ optimisation $\rightarrow$ hands-free operation — always inside safety guardrails
- Feasible with a 2–3-person team in 16–18 months; start with the Phase-1 prototype (~1.66M CNY)

This presentation was originally generated by OpenClaw.

Outlook

- Multi-facility “super digital twin”: federated learning, cross-site benchmarking
- Self-evolving twin: AutoML keeps models current; agents learn from every session
- Higher-fidelity real-time simulation: model-order reduction, GPU acceleration, quantum-assisted particle simulation
- Open standards & ecosystem: contribute to digital-twin standards, open-source core components

Thank you — Questions & Discussion

pchu@nju.edu.cn · Nanjing University