Data Orchestration and Large Model-Driven Interactive Diagnosis for LHAASO Operational Data

Minhao Gu, Huang Li, Xiao Chuan Xie, Hangchang Zhang, Ming Xuan Zhang
TIANFU Cosmic Ray Research Center, No.1500 Kezhi Road, Chengdu, China
Institute of High Energy Physics, Beijing 100049, China

Abstract

The Large High-Altitude Air Shower Observatory (LHAASO) generates massive multi-dimensional operational data in the course of continuous operation, posing prominent challenges to efficient data management and rapid analytical decision-making. To address these challenges, this paper proposes a comprehensive framework integrating data orchestration, intelligent processing, and interactive diagnosis. It classifies LHAASO data by defining the core concepts of "event" and "parameter," realizing the structured organization of heterogeneous data. A hybrid architecture combining persistent storage and stream processing is designed to ensure the reliable integration, high-capacity storage, and low-latency retrieval of TB-level data, with a minimum response time reaching the millisecond level. Based on this, a large model-driven intelligent analysis module is developed: users input natural language requirements via a web interface, and the system can automatically generate data processing code, execute tasks, and visualize results, effectively lowering the technical threshold for non-experts. Experiments demonstrate that the framework can process data efficiently, the large model can enhance analytical convenience, and the system can complete anomaly detection and rapid positioning within seconds, providing an intelligent solution for LHAASO and a reference for the management, analysis, and intelligent development of massive data in large scientific facilities.

01 Integrated Framework: From Data to Intelligent Decision-making

LHAASO Large Model-Driven Intelligent Analysis Workflow

Closed-loop feedback
State Awareness → Intelligent Analysis → Diagnostic Explanation → Feedback Optimization

1 Natural-language Request
Operators enter analysis goals, device objects, time ranges, or ab normal symptoms via the web interface

2 Web Interaction Entry
Unified entry for natural-language input, status snapshots, history queries, and visualization requests

3 Intent Understanding & Planning
The large model parses intent and determines data sources, proce ssing steps, and output formats

4 Data Retrieval & Context Building
Retrieve events, parameters, status snapshots, historical data, and knowledge-base records

5 Code Generation & Execution
Generate and schedule code for reading, cleaning, statistical analy sis, and plotting

6 Result Validation & Anomaly Detection
Check execution results, identify abnormal patterns, and infer syst em status and fault causes

7 Visualization & Diagnostic Explanation
Output charts, fault explanations, confidence levels, and recomme nded actions

8 Human-in-the-loop Feedback
Confirm, ask follow-up questions, tune parameters, or trigger the next analysis round

Diagnostic explanation / suggested actions are accumulated into the knowledge base

Supporting Capability Layer

A Data Governance & Structuring
Event / parameter models, unified heterogeneous-data descripti on, and metadata management

B Hybrid Storage & Stream Processing
Persistent storage, stream processing, TB-level data integration, and millisecond-level retrieval

C Knowledge Base & Historical Cases
Fault cases, expert rules, historical diagnosis records, and operati onal experience

D Safe Execution & Audit
Sandbox execution, logging, result tracing, and control-action c onfirmation

Data Foundation Events • Parameters • Status Snapshots • Historical Data • Knowledge Base

Outputs Execution Results • Charts • Anomaly Localization • Fault Causes • Recom mended Actions

02 Key Capabilities

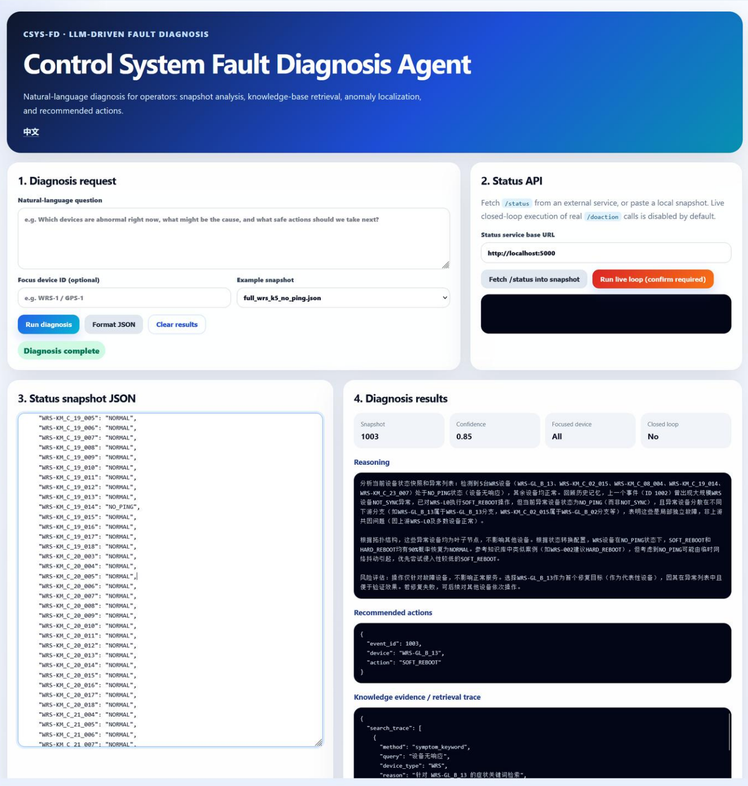
Structured Organization
Operational data are uniformly described with event and parameter concepts, integrating device status, operational events, configuration parameters, and analysis results into a shared semantic framework.

Hybrid Architecture
Persistent storage is combined with stream processing to support reliable integration, high-capacity storage, low-latency retrieval, and online analysis extension.

Large-model Analysis
Users describe analysis requirements in natural language; the system parses intent, generates processing code, executes tasks, and transforms results into visual feedback.

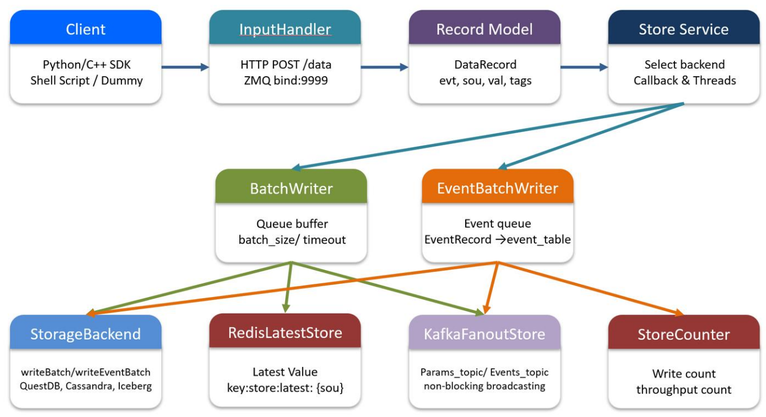
Interactive Diagnosis
Knowledge base, status snapshots, and agent reasoning are integrated for anomaly detection, fault explanation, action recommendation, and human-machine collaborative decision-making.

03 Interactive Web Console



The web console integrates natural-language diagnosis, status-snapshot editing, sample loading, API fetching, and result display, forming an interactive entry point for intelligent analysis.

04 Parameter Store Flow



Storage Pipeline
Clients write parameters through Python/C++ SDKs, shell scripts, or dummy data sources. HTTP / ZMQ input handlers parse DataRecord and EventRecord objects. StoreService assembles the StorageBackend according to application.conf, choosing QuestDB, Cassandra, or Iceberg. BatchWriter and EventBatchWriter aggregate parameter and event records by batch_size and timeout before persistence. Writes are also fanned out to RedisLatestStore to keep the latest value of each source, and to KafkaFanoutStore through params_topic / events_topic, forming three paths: persistence, cache, and message distribution.

05 LLM Intelligent Analysis Workflow

1 Natural-language Request
Operators describe analysis goals, device objects, or abnormal symptoms in natural language.

2 Task Planning
Identify intent, data sources, time ranges, and required processing steps.

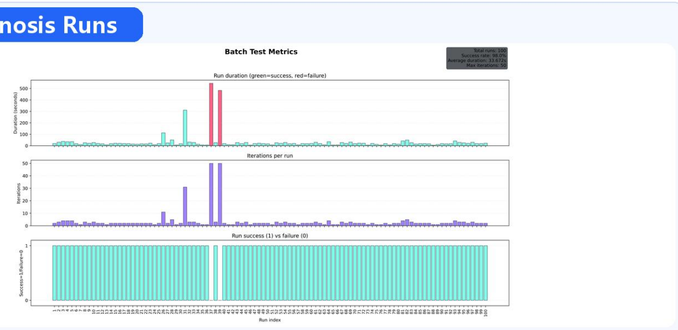
3 Code Generation
Automatically generate code for data reading, cleaning, statistics, and plotting.

4 Execution & Validation
Run tasks, capture exceptions, and check output consistency.

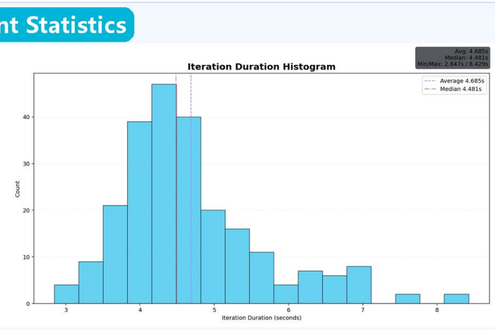
5 Visual Feedback
Return charts, diagnostic explanations, recommended actions, and traceable results.

06 Experimental Results and Demonstration

Batch Diagnosis Runs



Iteration-count Statistics



Experiment Description and Operational Conclusions

The system validates the agent analysis workflow through batch diagnosis tasks. After abnormal states and operational logs are provided, it automatically completes task planning, tool invocation, result summarization, and visual output. The side-by-side charts show batch-task results and iteration-count distributions, supporting observation of task success rate, convergence behavior, and anomaly-localization paths.

Efficient Processing Supports TB-level data integration, hybrid storage, and low-latency retrieval for continuous operational data management.

Lower Analysis Barrier Users describe analysis goals in natural language; the system generates code, executes tasks, and returns chart results.

Rapid Localization In typical fault scenarios, anomaly detection, cause explanation, and next-action suggestions are completed within seconds.

Traceable Closed Loop Records task inputs, iteration processes, tool outputs, and final judgments for review and continuous optimization.

Value and Broader Impact Natural-language Analysis · Interactive Diagnosis · Intelligent Operation

This work integrates data management, low-latency processing, natural-language analysis, and fault diagnosis into a reusable digital operation framework. Its value lies not only in improving the management efficiency of LHAASO operational data, but also in lowering the barrier to complex data analysis, enhancing the real-time capability, interpretability, and traceability of anomaly handling, and providing a reference for intelligent operation, remote collaboration, and long-term maintenance of large scientific facilities.