MIMOSA OSA-CBM DA 모듈의 현대화 방향

이현준, 박찬, 서채연, 김혜진, 임애린, 정남진, 원상운, 오은지*
*(주)에이아이네트웍스
e-mail:ai@ainetwork.co.kr

Modernization Direction of the MIMOSA OSA-CBM DA Module

Hyun-Jun LEE, Chan Park, Chae-Yeon Seo, Hye-Jin Kim, Ae-Rin Lim, Nam-Jin Jeong, Sang-Un Won, Eun-Ji Oh *,
*AI Networks Co..Ltd

요 약

MIMOSA OSA-CBM V3.1의 DA(Data Acquisition) 계층은 2000년대 초 소규모 유선 센서 환경, 주기적 폴링(Polling) 기반 수집 방식, 단일 관계형 데이터베이스(RDB) 저장 구조를 전제로 설계되었다. 그러나 현대 제조·플랜트 환경은 수백에서 수천 개의 IoT 센서가 초당 수십만 건의 시계열 데이터를 생성하는 구조로 전환되었으며, 기존 DA 계층은 데이터 적재 속도, 파형 저장 효율, 실시간 처리, 다중 프로토콜 호환성 측면에서 구조적 한계를 드러내고 있다.

본 논문은 기존 OSA-CBM DA 계층이 정의하는 7가지 데이터 타입과 이로부터 도출된 8가지 구조적 한계를 체계적으로 분석한다. 이를 바탕으로 Lambda·Kappa 아키텍처의 비교 분석을 수행하고, DA 계층의 실시간성 요구에 최적화된 Kappa 스트리밍 아키텍처를 채택한 CBM+ 현대화 설계를 제안한다. 핵심 제안으로는 폴링 방식에서 MQTT Push 방식으로의 전환, 엣지 게이트웨이 기반 적응형 샘플링, 동적 컨텍스트 기반 임계값 관리, 4종 신규 DA 타입 정의, PostgreSQL+TimescaleDB 하이브리드 데이터베이스 아키텍처를 포함한다. 제안 아키텍처는 파형 저장 I/O 99% 감소, 스토리지 70~90% 절감, 오경보 발생률 40~70% 감소, 네트워크 대역폭 90~99% 절감의 정량적 개선 효과가 기대된다.

1. 서론

설비 상태기반 정비(CBM)의 국제 표준인 MIMOSA OSA-CBM V3.1은 DA·DM·SD·HA·PA·AG 6계층 아키텍처를 정의한다. 이 중 DA 계층은 물리 센서로부터 원시 측정값을 수집하는 최하위 계층으로서, 전체 CBM 파이프라인의 데이터 품질을 결정하는 핵심 기반이다[1]. 해당 표준은 2000년대 초 소규모 유선 센서 환경, 주기적 폴링(Polling) 기반 수집 방식, 단일 관계형 데이터베이스(RDBMS) 저장 구조를 전제로 설계되었다.

그러나 현재 제조·플랜트 환경은 수백에서 수천 개의 IoT 센서가 초당 수십만 건의 시계열 데이터를 생성하는 고밀도 스트리밍 구조로 전환되었다. OPC-DA 기반 폴링 방식은 Linux·ARM 기반 엣지 환경에서의 운용이 사실상 불가능하며, 기존 단일 RDB 구조는 대규모 시계열 데이터의 적재·처리 과정에서 구조적 한계를 드러내고 있다[2]. 이러한 운용 환경의 변화는 DA 계층 전반에 대한 근본적인 재설계를 요구한다.

이에 본 논문은 OSA-CBM DA 계층이 정의하는 7가지 데이터 타입과 기존 RDB 기반 설계의 8가지 구조적 한계를 분석하고, Lambda·Kappa 아키텍처의 비교 검토를 통해 DA 계층에 최적화된 Kappa 스트리밍 아키텍처 기반의 CBM+ 현대화 설계를 제안하며, OSA-CBM의 의미론적 계층 구조와의 하위 호환성을 유지하는 점진적 현대화 전략을 지향한다[2].

2. 본론

2.1 기존 OSA-CBM DA 모듈 분석

2.1.1 기존 수집 구조와 데이터 흐름

기존 OSA-CBM DA 계층의 데이터 수집 구조는 DA 서버가 센서·장비에 주기적으로 데이터를 요청하는 폴링 방식으로 구성된다. 센서·장비는 OPC-DA 또는 Modbus TCP/RTU 프로토콜을 통해 고정 주기로 데이터를 제공하며, DA 서버는 ODBC/ADO를 통해 단일 RDBMS(MIMOSA DB)에 동기 방식으로 직접 기록한다. 수집된 데이터는 이후 상위 계층(DM→SD→HA→PA→AG)으로 순차적으로 전달된다[1].

2.1.2 구조적 한계 분석

기존 RDB 기반 DA 구조는 [표 1]과 같이 8가지 구조적 한계를 가진다[2].

[표 1] 기존 RDB 기반 DA 계층의 8가지 구조적 한계

구조적 한계	주요 내용
확장성 부족	센서 수 증가 시 INSERT TPS가 급격히 증가하여 RDBMS 처리 한계를 초과
파형 저장 비효율	DAWaveform은 1포인트당 1Row 저장 구조로 인해 대량 INSERT 발생
시계열 쿼리 비효율	시간 범위 조회 시 대규모 테이블 풀 스캔 발생
집계 성능 부재	평균·최댓값·RMS 계산 시 원시 데이터 전체 집계 필요
압축 부재	원시 데이터를 무압축으로 저장하여 스토리지 사용량 증가
실시간 처리 불가	배치 기반 처리 구조로 인해 이상 감지 반응 지연
프로토콜 제약	OPC-DA는 Windows COM/DCOM 기반으로 이기종 환경 지원이 제한됨
단일 장애점(SPOF)	단일 서버 장애 시 전체 데이터 수집 및 저장이 중단

기존 RDB 기반 DA 구조는 데이터 수집량 증가와 함께 저장·조회·집계 성능이 급격히 저하되며, 특히 고주파 파형 데이터 처리와 실시간 이상 감지 요구사항을 충족하기 어렵다. 또한, 플랫폼 종속성과 단일 장애점(SPOF) 문제는 대규모 CBM 환경에 적용하기에는 구조적 한계가 존재하며, DA 아키텍처의 재설계를 요구한다.

2.2 대규모 데이터 처리 아키텍처 패턴 비교

2.2.1 Lambda·Kappa 아키텍처 비교

Lambda 아키텍처는 Batch Layer·Speed Layer·Serving Layer의 3계층으로 구성되며, 정확성과 실시간성을 동시에 확보할 수 있다. 원본 데이터 보존 및 재처리가 가능하고 확장성이 높으나, Batch와 Stream 처리 로직을 이중으로 구현해야 하므로 개발·운영 부담이 크다[3].

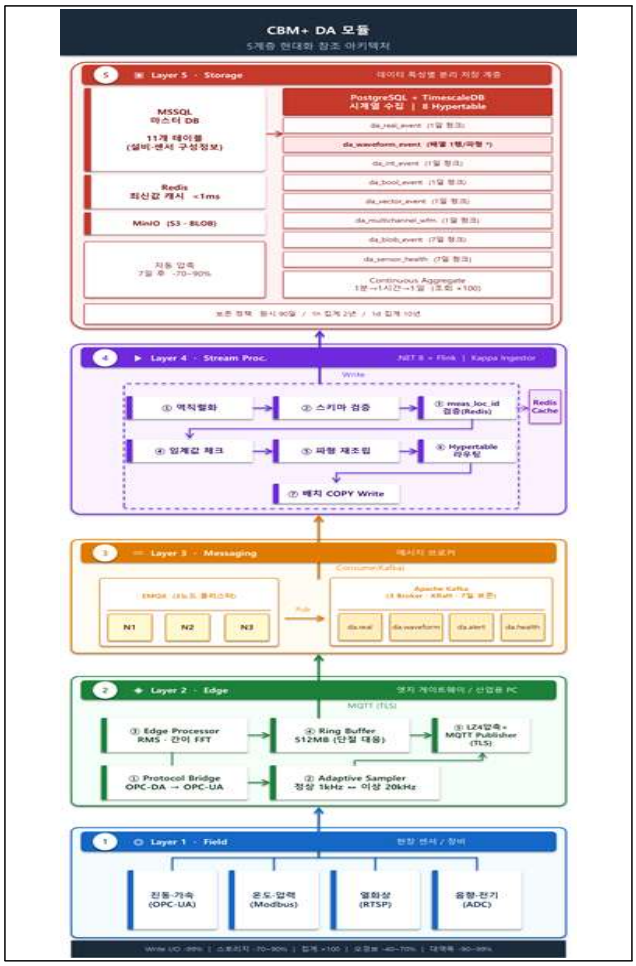
Kappa 아키텍처는 모든 데이터를 스트림 방식으로 처리하는 단일 파이프라인 구조로, Lambda의 이중 구현 문제를 해소한다. Message Broker(Kafka) 기반의 이벤트 스트림과 Stream Processing 엔진을 통해 실시간 처리에 최적화되어 있으며, 구조가 단순하여 유지보수가 용이하다. 다만 스트림 처리 시스템에 대한 기술 의존도가 높고, 대규모 과거 데이터 재처리 시 시스템 부하가 증가할 수 있다[3].

2.2.2 DA 계층 적용 아키텍처 선정

Lambda 아키텍처는 복잡한 분석·처리 단계가 포함된 DM~AG 상위 계층에 적합하다. 반면 DA 계층은 데이터 검증→임계값 체크→라우팅→저장의 단순한 처리 구조를 가지므로 Lambda의 이중 구현 복잡도를 도입할 필요성이 낮다. 따라서 DA 계층에는 구조가 단순하고 실시간 처리에 최적화된 Kappa 아키텍처를 채택한다. Kafka의 7일 보존 정책 내에서 재처리가 가능하여 DA 계층의 운영 요구사항을 충분히 충족한다[2,3].

2.2.3 CBM+ DA 5계층 현대화 참조 아키텍처

Kappa 아키텍처를 기반으로, CBM+ DA 모듈은 현장 데이터 수집부터 저장까지의 전 과정을 5개의 기능 계층으로 구조화한 참조 아키텍처를 정의한다. 각 계층은 Field Layer(현장 센서·설비), Edge Layer(엣지 게이트웨이), Messaging Layer(EMQX+Kafka), Stream Processing Layer (Flink 기반 스트림 처리), Storage Layer(PostgreSQL+TimescaleDB+Redis+MinIO)로 구성되며, 단방향 스트림 파이프라인으로 연결된다.



[그림 1] CBM+ DA 5계층 현대화 아키텍처

[그림 1]은 CBM+ DA 5계층 현대화 아키텍처의 전체 구조를 나타낸다. 본 아키텍처는 수집 방식 개선, 임계값 처리, DA 타입 확장 및 데이터베이스 설계의 구현 기반으로 활용된다.

2.3 데이터 수집 방식 현대화

2.3.1 폴링 방식에서 MQTT Push로의 전환

기존 OSA-CBM DA 계층은 서버가 센서·장비에 주기적으로 데이터를 요청하는 폴링(Pull) 방식을 사용한다. 이 방식은 구조가 단순하고 구현이 용이하나, 센서·장비 수가 증가할수록 서버의 네트워크 트래픽과 시스템 부하가 급증하며, 이벤트 기반 실시간 처리에 한계가 있다. CBM+ 현대화에서는 센서·

장비가 MQTT Broker에 데이터를 직접 발행(Publish)하는 Push 방식을 적용한다. MQTT QoS-1을 통해 메시지 전달 신뢰성을 보장하며, 불필요한 폴링 요청을 제거하여 네트워크 효율성을 높이고 대량의 IoT 센서 데이터를 실시간으로 수집·처리할 수 있다[2,4].

2.3.2 엣지 전처리와 적응형 샘플링

기존 구조에서는 센서 데이터를 중앙 서버로 직접 전송하는 중앙 집중형 처리 방식을 사용한다. 이 방식은 센서 수가 증가할수록 네트워크 트래픽이 급격히 증가하고 중앙 서버의 처리 부하가 커지는 문제가 발생한다. CBM+에서는 엣지 게이트웨이에서 1차 데이터 전처리를 수행하는 분산 처리 구조를 적용한다. 엣지에서는 RMS 계산·간단한 이상 탐지 등 경량 분석을 수행하며, 정상 상태에서는 평균·최대·최소·RMS 등 요약값만 전송하고 이상 징후 감지 시에만 원시 파형 데이터를 전송한다. 적응형 샘플링은 정상 운전 시 낮은 샘플링 주파수(1kHz), 이상 탐지 시 높은 샘플링 주파수(20kHz)로 자동 전환하여 이상 발생 전후의 고해상도 데이터를 확보한다. 이를 통해 네트워크 대역폭을 90~99% 절감할 수 있다[2].

2.3.3 동적 컨텍스트 기반 임계값 관리

기존 CBM에서는 고정된 min/max 임계값을 기준으로 이상 여부를 판단한다. 동일한 임계값이 모든 운전 조건에 적용되므로, 저속·저부하 운전 등 특정 조건에서 정상 데이터가 경보로 판단되는 오경보가 빈번히 발생한다. CBM+에서는 설비의 회전 속도, 부하율, 운전 상태 등 운전 컨텍스트를 수집 데이터와 함께 operating_speed·load_percent·ambient_temp 컬럼에 저장하고, ML 모델이 동일 운전 조건의 과거 데이터를 기반으로 $\mu \pm 3\sigma$ 동적 임계값을 실시간으로 산출한다. 이를 통해 설비별 개인화된 기준선을 학습하여 오경보를 40~70% 감소시키고, 이상 패턴의 조기 감지가 가능해진다[2].

2.4 CBM+ 신규 데이터 타입 정의

2.4.1 신규 데이터 타입 도입 배경

기존 7가지 OSA-CBM DA 타입은 단일 채널·단일 센서·고정 샘플링 구조를 전제로 설계되어, 현대 IoT 환경의 다축 센서, 적응형 수집, 대용량 스트리밍, 센서 건강 관리 요구사항을 충족하지 못한다. CBM+ 현대화에서는 이러한 한계를 보완하기 위해 [표 2]와 같이 4종의 신규 DA 타입을 정의한다[1,2].

DAMultiChannelWaveform은 다축 센서를 동일 클럭으로 동기화하여 단일 패킷으로 전송하는 타입으로, JOIN 없이 채널 간 위상 동기화 및 상관 분석이 가능하다. DASensorHealth는 센서 자체 건강 진단 타입으로, SNR·잔여 수명(RUL)·드리프트율·교체 권고 여부 등을 포함하여 센서 PHM을 지원한다. DAContextData는 설비의 운전 컨텍스트 변화 이벤트를 수집하는 타입으로, 운전 모드·회전 속도·부하율·외기 온도·변경 사유 등의 필드를 포함한다. DAStreamSegment는 대용량 파형 데이터를 청크 단위로 분할 전송하고 시퀀스를 관리하여 재조립 후 저장하는 스트리밍 특화 타입이다[2].

[표 2] CBM+ 신규 DA 타입 및 저장 구조

신규 타입	목적	저장 위치
DAMultiChannel Waveform	3축 동기 파형, 위상 분석	da_multichannel_wave form
DASensorHealth	센서 건강 진단 (SNR·RUL)	da_sensor_health_event
DAContextData	운전 컨텍스트 변화 이벤트	da_context_event
DAStreamSegment	대용량 파형 청크 스트리밍	재조립→da_waveform_ event

2.5 RDB+TSDB 하이브리드 데이터베이스 설계

2.5.1 단일 RDB 구조의 한계와 분리 전략

IoT 센서 데이터는 본질적으로 두 가지 특성의 데이터로 구성된다. 설비 정보·센서 위치·수집 설정·경보 기준 등 구성·마스터 데이터는 변경 빈도가 낮고 복잡한 JOIN·FK 제약·트랜잭션 무결성이 필요하다. 반면 시계열 수집 데이터(DA*Event)는 초당 수천~수만 건이 발생하며, 시간 범위 조회·집계가 주요 쿼리 패턴이고 수평 확장이 필수적이다. 기존 OSA-CBM은 이를 구분하지 않고 단일 RDBMS에 저장함으로써 대규모 센서 데이터 처리에 구조적 한계가 발생한다. CBM+에서는 PostgreSQL(구성·마스터)과 TimescaleDB(시계열 수집)로 저장소를 분리하고, meas_loc_id를 연결 키로 사용하여 논리적 정합성을 유지한다[2].

2.5.2 PostgreSQL 마스터 DB 테이블

PostgreSQL 마스터 DB는 site(사이트 식별), asset(설비 정보), segment(설비 세그먼트), measurement_location(측정 위치·TSDB 연결 키), da_out_port(DA 채널 설정), eng_unit(공학 단위 마스터), alert_type(경보 타입), alert_severity_type(경보 심각도), alert_region(경보 구역 설정), transducer_type(트랜스듀서 타입), alert_region_ref(경보 구역 참조)의 11개 테이블로 구성된다. ACID 트랜잭션, FK 제약, JOIN 최적화를 통해 설비 구성 정보의 무결성을 보장한다[2].

2.5.3 TimescaleDB Hypertable 시계열 테이블

TimescaleDB는 PostgreSQL 기반으로 시간 기준 자동 파티셔닝(청크 분할)을 제공하며, 압축·연속 집계·보존 정책을 기본으로 내장한다. CBM+ DA 현대화에서 정의하는 8개의 Hypertable은 [표 3]과 같다.

[표 3] TSDB Hypertable

테이블 명	DA타입/청크간격	주요 컬럼
da_real_event	DAReal/1일	time,meas_loc_id,value, confid,operating_speed, load_percent, snr_db
da_waveform_event	DAWaveform/1일	time, sample_count, meas_loc_id,valuesDOUBLE PRECISION[],x_axis_delta
da_int_event	DAInt/1일	time, meas_loc_id, value,data_status
da_bool_event	DABool/1일	time, meas_loc_id,value, previous_value
da_vector_event	DAVector/1일	time, meas_loc_id,values[], x_delta
da_multichannel_ waveform	DAMultiChannel Waveform/1일	time, meas_loc_id,channel_count, channel_ids[],channel_data[]
da_blob_event	DABLOBData/7일	time,meas_loc_id,content_type ,data_ref (MinIO Key)
da_sensor_health_ event	CBM+ 확장/7일	time,meas_loc_id,health_sco re,snr_db,fault_type, sensor_rul_hours

2.6 종합 성능 비교 및 기대 효과

2.6.1 기존 방식 대비 정량적 개선 효과

CBM+ DA 현대화 아키텍처는 기존 OSA-CBM V3.1 구조 대비 [표 4]와 같은 정량적 성능 개선을 달성한다.

[표 4] 기존 OSA-CBM 대비 CBM+ DA 현대화 아키텍처 성능 비교

항목	기존 OSA-CBM	CBM+ 현대화	개선 효과
수집 방식	고정 주기 폴링	MQTT Push 적응형	수집 지연 ms 단위로 감소
수집 프로토콜	OPC-DA (COM/DCOM)	OPC-UA+MQTT TLS	IoT 보안 표준 준수, Linux·ARM 지원
저장 DB	PostgreSQL 단일 RDB	PostgreSQL+Time scaleDB 하이브리드	Write 성능 10배 이상 향상
파형 저장	포인트당 1행	배열 1행/파형	I/O 99% 감소
임계값 관리	고정 min/max	동적 ML 기반 컨텍스트 인식	오경보 40~70% 감소
집계 처리	원시 풀스캔	Continuous Aggregate	조회 속도 100배 향상
데이터 압축	없음	TimescaleDB 자동 압축	스토리지 70~90% 절감
보존 관리	수동 DELETE	자동 보존·삭제 정책	운영 부담 최소화
엣지 처리	없음	엣지 이상 탐지+적응형 전송	대역폭 90~99% 절감
스케일링	수직 확장	수평 확장 브로커 클러스터	수만 센서 동시 지원
고가용성	없음(SPOF)	전 계층 다중화	단일 장애점 제거

2.6.2 AI/ML 학습 데이터 품질 향상

CBM+ DA 현대화 구조는 AI/ML 기반 예지 분석을 위한 풍부한 학습 데이터를 기본으로 제공한다. 운전 컨텍스트(RPM·부하율·주변 온도)가 수집 데이터와 동시에 저장되어 동일 운전 조건 기반의 정밀한 이상 탐지 모델 학습이 가능해진다. SNR·completeness 기반 데이터 품질 가중치 적용으로 불량 학습 데이터가 자동 필터링되며, 센서 건강 정보(DASensorHealth)를 통해 센서 PHM 모델의 학습 데이터도 자동으로 축적된다. 이를 통해 기존 CBM의 단순 스칼라 기반 분석에서 파형·컨텍스트·품질·건전성 등 다차원 피처를 활용하는 고도화된 예지 분석으로 발전할 수 있다[1,2].

3. 결론

본 논문은 MIMOSA OSA-CBM V3.1 DA 계층의 7가지 데이터 타입과 8가지 구조적 한계를 분석하고, IoT·엣지 컴퓨팅·TimescaleDB·Kappa 스트리밍 아키텍처를 통합한 CBM+ DA 현대화 설계를 제시하였다. Lambda·Kappa 아키텍처 비교를 통해 DA 계층의 단순한 처리 특성에 최적화된 Kappa 방식을 채택하였으며, DM~AG 상위 계층과의 역할 분리를 명확히 하였다.

제안 설계의 핵심 기여는 다음과 같다. OSA-CBM V3.1의 의미론적 계층 구조를 유지하는 하위 호환적 현대화 전략을 제시하였으며, MQTT Push·엣지 적응형 샘플링·동적 ML 임계값·PostgreSQL+TimescaleDB 하이브리드 구조를 통해 대역폭 90~99% 절감, 오경보 40~70% 감소, I/O 99% 감소, 집계 조회 100배 향상의 정량적 성과를 달성하였다. 또한 4종 신규 DA 타입 정의를 통해 다축 센서·센서 PHM·운전 컨텍스트 관리 기반을 마련하였다.

이를 통해 CBM 표준 아키텍처의 현대화 방향을 제시하고, 산업 IoT 환경에서의 실용적 적용 가능성을 확인하였다는 점에서 의의를 가진다. 향후 연구로는 실증 환경에서의 파일럿 성능 검증, DM~AG 계층으로의 CBM+ 확장 설계, 디지털 트윈과의 통합 방안이 요구된다.

사사

본 논문은 과학기술정보통신부 소형모듈원자로 디지털 혁신 검증·운영 기술개발 사업의 지원으로 작성되었음. [RS-2025-02316385]

참고문헌

[1] MIMOSA, "OSA-CBM: Open System Architecture for Condition-Based Maintenance, Version 3.1," MIMOSA Standard, 2010.

[2] 소형모듈원자로 핵심 기기 상태감시 및 혁신 유지보수 기반 기술 개발 과제, "CBM+ DA 모듈 설계서 — MIMOSA OSA-CBM 기반 IoT·CBM+ 기술 적용 현대화 보고서," 내부 기술 문서, 2026.

[3] J. Kreps, "Questioning the Lambda Architecture," O'Reilly Media, 2014.

[4] OASIS, "MQTT Version 5.0," OASIS Standard, 2019; OPC Foundation, "OPC Unified Architecture, Part 14: PubSub," 2018.