

경력기술서_문성준

DBA & Backend Engineer

작성일: 2026.06.23

경력 요약

제조 ERP/MES 플랫폼 개발 프로젝트에서 DBA와 백엔드 개발을 함께 담당했습니다. Oracle 기반 레거시 시스템을 PostgreSQL 기반 구조로 전환하고, Spring Boot/MyBatis 기반 업무 API, 복잡한 ERP SQL, DB 형상관리, 배포 자동화, 모니터링, 백업/복구 전략까지 개발과 운영이 이어지는 범위를 맡았습니다.

특히 데이터베이스 구조와 업무 프로시저를 기준으로 하면, API, SQL이 같은 규칙으로 동작하도록 구현했습니다. 기능 개발뿐 아니라 데이터 정합성, 반복 가능한 배포, 장애 원인 추적, 문서화와 인수인계까지 고려해 작업했습니다.

주요 기술 역량

- PostgreSQL 기반 DB 설계, Oracle to PostgreSQL 마이그레이션, PL/pgSQL 변환
- Spring Boot, MyBatis, FreeMarker 기반 ERP 백엔드 API 및 동적 SQL 개발
- Liquibase 기반 DB 변경 이력 관리와 dev/qa/main 환경 반복 배포
- Jenkins, Ansible, Docker 기반 배포 자동화와 서버/환경 이관
- Grafana, OpenTelemetry, Loki, Tempo 기반 서비스/DB/인프라 관측성 구축
- AI, RAG, MCP 기반 업무 문서 검색 및 ERP 실시간 데이터 연동
- Claude, Codex를 활용한 소스 조사, 코드 리뷰, 문서 작성 업무 개선

주요 성과

- Oracle DB 객체 전환 대상 관리: 테이블 180개, 함수 60개, 프로시저 101개, 뷰 5개
- DB 형상관리: Liquibase 변경 단위 500개 이상 구성 및 환경별 반복 배포 기준 마련
- CI/CD 구성: Backend, Frontend, DB, LLM 4개 배포 파이프라인 구축
- Ansible 자동화: PostgreSQL, Backend, 모니터링 에이전트 등 6개 role 구성
- 모니터링: Grafana 대시보드 7개, 운영 알림 규칙 24개, Teams 알림 연동
- AI/RAG: 내부 50문항 평가 기준 문서 검색 성공률 22%에서 66%로 개선

경력사항

런업컴퍼니 / DynaMOS 제조 ERP/MES 플랫폼

- 기간: 2025.11 ~ 현재
- 직무: DBA / Backend Engineer
- 팀 규모: 개발자 4~5명
- 담당 영역: DB 마이그레이션, 백엔드 API, ERP SQL, 배포 자동화, 운영 모니터링, AI/RAG
- 주요 기술: PostgreSQL, Oracle, Spring Boot, Java, MyBatis, Liquibase, Docker, Jenkins, Ansible, Grafana, FastAPI

DynaMOS는 제조업 공장을 대상으로 하는 ERP/MES 플랫폼입니다. 기준정보, 수주, 구매, 공정, 출하, 품질, 마감, 관리자 기능을 포함하며, 기존 Oracle 기반 구조를 PostgreSQL 기반 신규 구조로 전환하는 과정에서 DB, 백엔드, 배포, 모니터링, AI 기능을 함께 구현했습니다.

주요 수행 업무

1. Oracle to PostgreSQL 마이그레이션 및 DB 운영

- Oracle 기반 ERP/MES 레거시 DB를 PostgreSQL 17 기반 구조로 전환했습니다.
- 테이블, 함수, 프로시저, 뷰, 시퀀스 등 DB 객체를 분류하고 PostgreSQL 반영 순서와 Liquibase 변경 단위를 구성했습니다.
- PL/SQL 프로시저의 커서, 예외 처리, 날짜/문자열/null 처리 로직을 PL/pgSQL과 PostgreSQL 함수 기준으로 변환했습니다.
- Oracle 전용 JDBC, NLS, SQL 문법을 PostgreSQL JDBC, currentSchema, 환경별 datasource, MyBatis XML 기준으로 전환했습니다.
- dev, qa, main 환경별 DB와 LLM DB 백업 기준을 나누고, pg_dump, pg_dumpall, WAL/PITR, pgBackRest 적용 방향과 복구 절차를 문서화했습니다.

2. ERP 백엔드 API 및 동적 SQL 개발

- Spring Boot와 MyBatis 기반으로 기준정보, 수주/계획, 구매, 공정, 출하, 품질, 마감, 관리자 화면의 백엔드 API와 XML SQL을 개발했습니다.
- 출하현황, 원자재불출, 순수요계획, 월수불마감, 작업지시, 공정검사, 재고현황 등 주요 화면의 조회/저장 SQL을 PostgreSQL 기준으로 수정했습니다.
- MyBatis 정적 SQL만으로 처리하기 어려운 일별, 주별, 월별 pivot 컬럼과 합계 계산을 FreeMarker 기반 동적 SQL로 전환했습니다.
- 수요계획, 생산계획, 외주계획, 자재소요계획, 정규발주 등 계획 계열 화면에서 기간, 버킷 유형, 소개, 합계를 서버 기준으로 처리했습니다.
- 다국어, 메뉴, 권한, 그리드 컬럼 등 관리자 기능은 프론트 하드코딩을 줄이고 DB 데이터를 기준으로 동작하도록 개선했습니다.

3. CI/CD, 서버 이관, 모니터링 구축

- Jenkins 기반 Backend, Frontend, DB, LLM 배포 파이프라인을 구성하고 브랜치 기준으로 개발, QA, 운영 환경이 매핑되도록 구현했습니다.
- Ansible을 활용해 PostgreSQL 설치, 백엔드 배포, Node Exporter, PostgreSQL Exporter, OpenTelemetry 에이전트 배포를 자동화했습니다.
- 기존 단일 서버 중심 구조에서 DB, LLM, Jenkins, 모니터링 역할을 분리하는 서버 이관을 수행했습니다.
- 설정 공유 방식으로 submodule을 검토했지만, 운영 결합도를 낮추기 위해 별도 configuration repository와 OpenBao 기반 민감정보 관리 구조를 채택했습니다.
- Grafana, OpenTelemetry, Mimir, Loki, Tempo를 연동해 서비스, DB, 인프라 상태를 대시보드와 Teams 알림으로 확인할 수 있게 구축했습니다.

4. AI 기능 및 ERP 데이터 연동

- FastAPI 기반 LLM 서비스에서 질문 분류, 문서 검색, ERP API 호출, 응답 스트리밍 기능을 구현했습니다.
- PDF, Markdown 등 업무 문서를 RAG 데이터로 활용하기 위해 파서, 청킹 기준, 메타데이터 저장 구조, 임베딩 저장 흐름을 구성했습니다.
- pgvector와 키워드 검색을 함께 활용하고, reranking과 캐시 전략을 적용해 검색 품질과 응답 속도를 개선했습니다.
- MCP를 통해 출하, 생산, 마감, 재고 등 ERP 실시간 데이터를 AI 채팅에서 조회할 수 있도록 백엔드 API 연동 구조를 구현했습니다.
- 임베딩 모델, 청킹 크기, overlap, 문서 구조 유지 여부를 비교할 수 있는 테스트 질문셋과 개선 전후 비교 절차를 마련했습니다.

5. 사용자 가이드 및 AI 도구 활용

- 기준정보, 수주, 공정, 출하, 품질, 마감 등 주요 MES 화면의 사용자 가이드와 업무 설명 문서를 작성했습니다.
- 화면별 조회조건, 입력 항목, 버튼 동작, 선행 조건, 관련 화면 흐름을 문서화해 팀에서 참고할 수 있게 했습니다.
- Claude와 Codex를 요구사항 정리, 소스 조사, 구현, 리뷰, 검증 단계로 나누어 활용했습니다.
- 작성자, 리뷰어, 테스트 실행 관점의 하네스를 정의하고 반복 작업용 검증 체크리스트로 표준화했습니다.
- 파일 목록과 검색 결과로 작업 범위를 먼저 좁히고 필요한 원본만 확인해 입력 범위와 추정 기반 작업을 줄였습니다.
- PR 리뷰, 장애 조사, DB 점검, 서버 이관, 사용자 가이드 작성 절차를 템플릿화해 재작업과 인수인계 비용을 낮췄습니다.
- 운영 서버, DB 변경, 민감정보 취급, Git 원격 반영 기준을 작업 규칙에 포함해 AI 도구를 안전하게 활용했습니다.

기술 스택

- Database: PostgreSQL, Oracle, Liquibase, pgBackRest, pgvector, PL/pgSQL
- Backend: Java, Spring Boot, MyBatis, FreeMarker, REST API
- DevOps: Jenkins, Ansible, Docker, Docker Compose, Linux, Shell Script
- Monitoring: Grafana, OpenTelemetry, Prometheus Exporter, Loki, Tempo
- AI/LLM: FastAPI, RAG, MCP, Ollama, Vector Search, Embedding, Reranking
- Frontend 경험: Next.js, React, TypeScript, AG Grid