

실패를 격리하고, 중단된 처리를 이어 가며, 틀어진 데이터를 바로잡았습니다

Java·Kotlin·Spring으로 실패 후 다시 시작할 수 있고
데이터 정합성을 지키는 백엔드를 개발합니다.
이벤트 복구, JPA 데이터 보정, 포인트 배치·동시성,
검색·캐시 설계의 네 가지 사례와 공개한 코드·글·발표를
정리했습니다.

서민재 · Rex Seo

Backend Engineer · Java / Kotlin /
Spring
ckdekrn88@gmail.com

GitHub

Velog

LinkedIn

이력서

경력기술서

01

실패 단계를 기록해 필요한 업무만 다시 실행했습니다

담당	구현	저장	알림
이벤트 상태 기록·수동 복구·정체 알림 경로 설계·개발	Kafka 이벤트의 발행·처리 상태를 단계별로 구분	PostgreSQL에 처리 상태 기록 · JSONB에 복구용 payload 저장	정체된 IN_PROGRESS 이벤트를 유형별로 묶어 Slack 알림
상황 같은 실패 상태로는 복구 지점을 정할 수 없었습니다 발행 전에 멈췄는지, 발행 후 업무 처리에서 멈췄는지에 따라 확인할 데이터와 재실행 범위가 달랐지만 하나의 실패 상태만 남았습니다.		바꾼 점 실패 단계와 업무를 분리해 복구 범위를 좁혔습니다 발행 실패와 처리 실패를 나눠 기록하고 payload를 보존했습니다. 포인트 차감·회수·만료 알림 가운데 완료되지 않은 업무만 골라 다시 실행했습니다.	

실제 영속성 컨텍스트에서 JPA 변경 감지 오류를 재현하고 약 200만 건을 보정했습니다

Mock 테스트에서 놓친 저장 오류를 실제 영속성 컨텍스트에서 재현했습니다. 보정 스크립트로 잘못 저장된 값을 바로잡아 저장 데이터를 다시 신뢰할 수 있게 했습니다.

문제	재현	보정	회귀 검증
JPA 변경 감지 오류	실제 영속성 컨텍스트에서 저장 과정 검증	오류 데이터 식별 · 보정 스크립트 적용	실제 저장 동작을 확인하는 테스트 보강

상황

Mock 테스트만으로는 실제 저장 오류를 찾을 수 없었습니다

기존 테스트는 실제 영속성 컨텍스트의 변경 감지와 저장 시점을 거치지 않아 오류가 드러나지 않았습니다.

바꾼 점

실제 저장 환경에서 재현한 뒤 보정 경로를 만들었습니다

오류 원인을 재현해 확인하고 보정 스크립트를 작성했습니다. 잘못 저장된 값을 바로잡아 저장 데이터를 다시 신뢰할 수 있게 했습니다.

중단된 배치를 이어 처리하고 포인트 정합성을 검증했습니다

배치마다 재시작 기준을 분리하고, 동시 요청이 끝난 뒤 잔액·원장·예산을 함께 대조했습니다.

담당	처리 단위	보관 대상	동시성
포인트 소멸·아카이브 배치 구현 · 만료 알림 개선	소멸 대상을 페이지 단위로 읽고 반복 처리	참조 순서대로 옮긴 아카이브 테이블 10개	9개 API · 13개 시나리오 · 3·5·10개 동시 요청

상황

배치마다 중단 후 이어갈 지점이 달랐습니다

소멸 배치는 잔액·원장을 바꾸고 만료 알림 배치는 발송 대상을 모았습니다. 아카이브 배치는 참조 순서를 지켜 테이블을 옮겨야 했습니다.

바꾼 점

진행 위치를 따로 기록하고 결과 정합성을 확인했습니다

배치별 진행 위치를 분리해 재실행 범위를 제한했습니다. 동시 요청 후에는 잔액·원장·예산·후속 이벤트를 함께 대조했습니다.

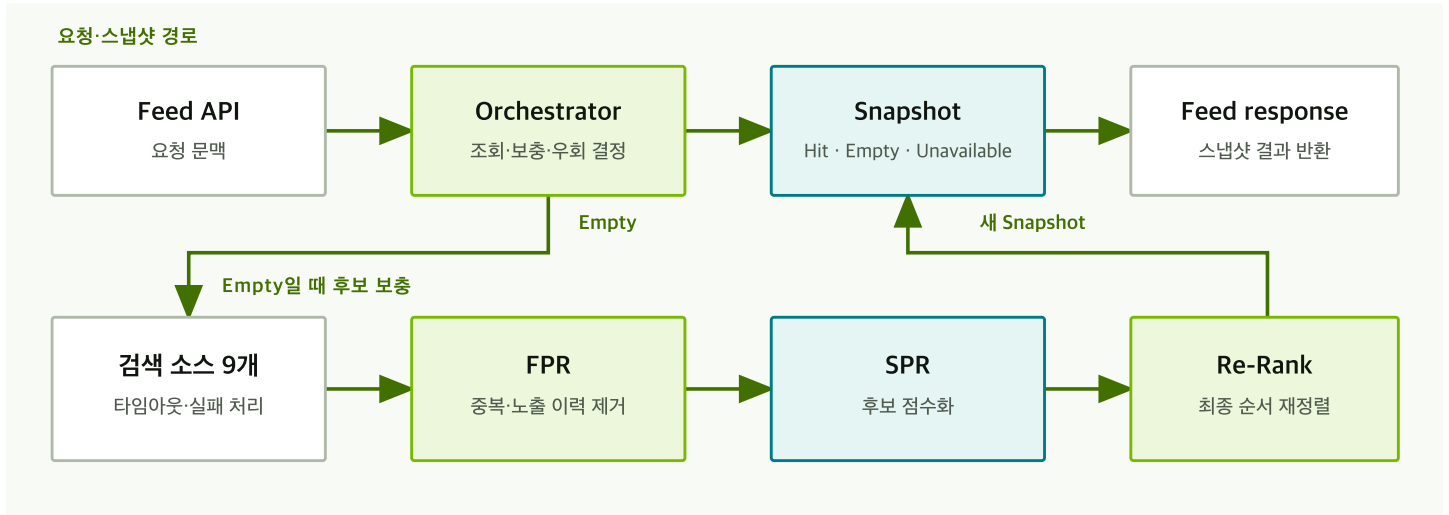
검색 소스의 실패를 격리하고 캐시 우회 경로를 만들었습니다

검색 소스별 정책·타임아웃·실패 처리를 분리했습니다. 캐시를 쓰지 않고 다시 계산하는 경로도 마련했습니다.

처리 단계	구성	실패 격리	연동
후보 조회 → 중복 제거 → 점수화 → 재정렬 → 캐시 저장	후보 수집 인터페이스 · 역방향 조회 방식 · 캐시 전략 분리	검색 소스별 정책·타임아웃·실패 처리	OpenSearch 조회 · Redis 캐시 우회 · 순환 의존 검출

후보를 모아 피드로 내보내기까지

Snapshot 상태에 따라 캐시를 쓰거나 랭킹을 다시 실행합니다.



발표·글·코드로 남긴 공개 기록

기술 글 156편과 YouthCon 발표, Spring 오픈소스 개선 제안, 개인 프로젝트를 한곳에 모았습니다.

YouthCon 2025 발표 '우리팀을 위한 Spring Test 만들기' 발표에서 반복되는 테스트 준비 코드를 공통화하고, 테스트 유형에 맞는 설정과 작성 기준을 정리한 과정을 다뤘습니다.	기술 글 156편 · 2026.07 기준 Java·JPA·테스트 관련 문제와 SQL 튜닝 경험을 글로 기록했습니다. 글또 10기에서는 준비위원과 연사로 참여했습니다.
Spring 오픈소스 · 반영된 개선 제안 Spring Kafka, Spring Boot, Spring Data JPA에 불변 컬렉션 처리, 하드코딩 제거, 자원 정리에 관한 개선을 제안했고 모두 반영되었습니다.	Java Cafe 기반 OpenSearch 한글 플러그인 Java Cafe Elasticsearch 플러그인을 OpenSearch 2.19에서 동작하도록 Kotlin으로 이식했습니다. 저장소에는 한·영 자판 변환과 초성·자모 검색을 다루는 assertion 예시 16개가 있습니다.
약 7만 건의 삭제 상태 불일치 파일 삭제 상태와 DB 기록이 어긋난 원인을 분석해 글로 남겼습니다.	JPA 오류 재현과 약 200만 건 보정 Mock 테스트에서 드러나지 않던 JPA 변경 감지 오류를 재현하고 보정한 과정을 글로 정리했습니다.