

경계를 넘는 에이전트

분리된 시스템, 연결된 판단

제조 · 금융 · 보험 - A2A 멀티홉 연동
독립 시스템 3사 · 서로를 모르는 세 에이전트의 협업 실험

발표자 : 최지현

MAINTQ

설비보전 에이전트

FINALLQ

은행 · 증권 에이전트

INSUQ

보험약관 RAG 에이전트

NIPA NVIDIA GPU 가속 기반
AI Agent 엔지니어 과정

2026.08.25

목차

01 - 02

문제 정의

은행·보험·설비를 따로 확인해야 하는 지금의 업무 구조

03 - 04

프로젝트 개요 · 전제

독립된 세 회사를 전제로 한 설계, 그리고 세 가지 가정

05 - 06

시나리오 지도

구현한 것과 곧 구현할 것을 나눠서

07 - 11

MaintQ - 설비보전

진단 · 발주 · 안전 · 자산판단, draft-only 쓰기 경계

12 - 16

InsuQ - 보험약관 RAG

근거 인용, 거부할 줄 아는 능력, 52개 실험

17 - 20

FinAllQ - 은행 · 증권

결재 라인, 이상거래 탐지, 직무 분리

21 - 24

프로젝트 간 통신(A2A 연동)

A2A 계약, 멀티홉 릴레이, 실행 결과

25 - 32

성과 · 한계 · 부록

성과
한계
부록

세 곳에 흩어진 판단 근거

현장

설비 구동 중단

에러코드 발생 → 매뉴얼 100+p → 재고/대체품 파악

에러코드는 뒀는데 매뉴얼은 PDF 수백 장.

이 부품이 재고에 있는지, 대체품이 되는지 각각 파악해야 함

자금

은행 상담 필요

조달 방식 · 한도 · 담보 가능성 - 별도 창구·별도 절차

교체 비용을 어떻게 조달할지,
한도가 남았는지, 담보가 되는지
별도 창구·별도 절차.

보장

약관 해독 난이도

부보 여부 · 보장 충분성 - 조·항·호 직접 탐색

이 설비가 보험에 들어 있는지, 보장이 충분한지.
조·항·호로 쓰인 문서를 사람이 직접 찾아야 한다.

연결 방식 = 담당자의 수기 이관 하나의 의사결정에 3곳의 결정이 필요합니다.

지연

판단 1건당 왕복 3회

누락

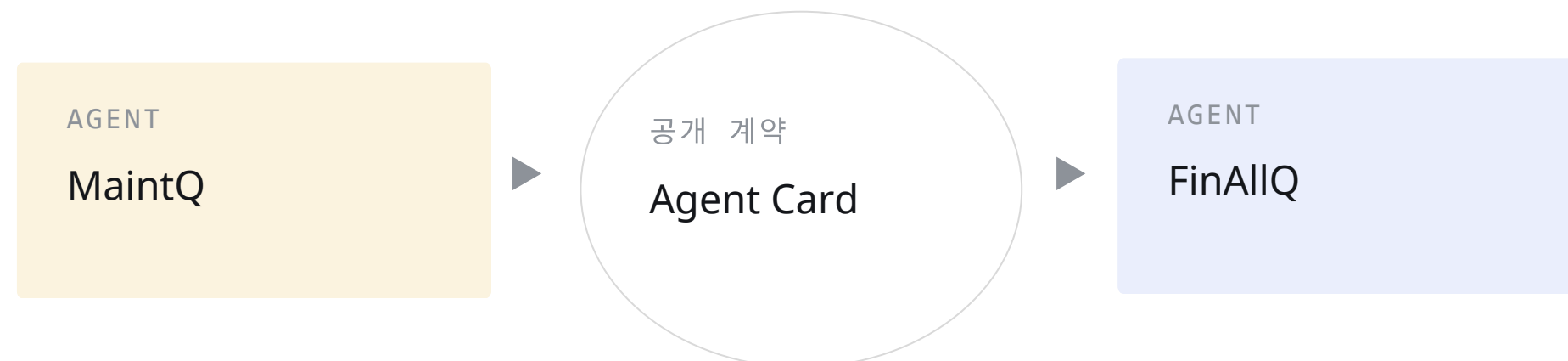
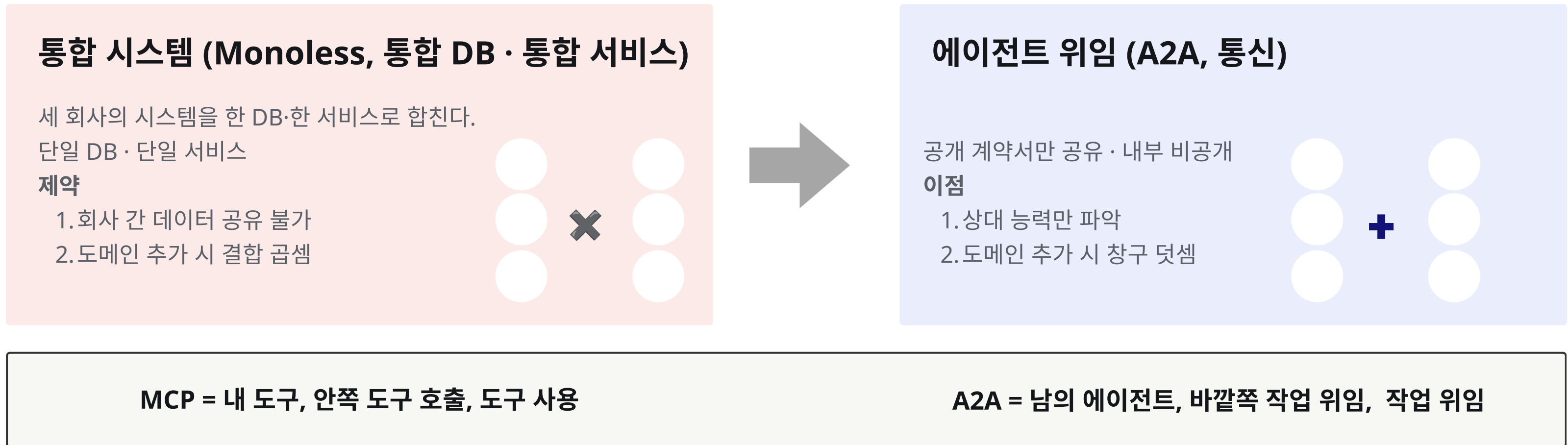
보험 통지 누락 → 청구 거절

추적 불가

판단 주체·시점·근거 미기록

모놀리스 vs A2A

현실에서는 통합은 불가능합니다. 은행은 공장 설비 DB를 볼 수 없고, 보험사는 고객 계약 원장을 통째로 넘겨주지 않습니다.



A2A로 세 개의 독립된 회사를 연결한다면

한 회사의 여러 서비스가 아닙니다. 서로의 코드를 수정할 수 없고, 상대가 죽어도 내가 살아야 하는 관계입니다.

가정 01

상호 내부 비공개

도메인 코드는 서로 손대지 않습니다. 접촉면은 HTTP와 Agent Card뿐. 스택도 일부러 다르게 뒀습니다.
벡터 스토어도 Qdrant와 pgvector로 갈립니다.

가정 02

최종 판단은 사람

에이전트 = 초안까지
자금·계약 지점에 승인 계단 · 자동 승인 경로 부재

가정 03

모르면 확인 불가

근거 없으면 추측 금지. 「확인 불가」를 회신합니다.
오답 회신이 최대 위험

3

독립 레포 · 독립 배포

0

서로의 DB 직접 접근

STP 전략

실사용자 특정

SEGMENT - 나누기

중견 제조업 설비 담당

설비 자산 다수 · 전담 재무/법무 부재

대기업은 이미 ERP로 묶여 있고,
소기업은 자산 규모가 작습니다.

TARGET - 고르기

정비팀장 · 재무 담당자

판단 책임 보유 · 정보 분산 · 실제 결재 주체

판단의 책임을 지지만 정보는 세 곳에 흩어져 있는
사람. 실제로 승인 버튼을 누르는 사람입니다.

POSITION - 자리잡기

대체가 아니라 준비

판단 대행 × · 결재 직전까지 근거 조립 ○

"판단을 대신해 주는 AI"가 아니라
"근거를 모아 결재 직전까지 준비해 주는 AI".
결정권은 넘기지 않습니다.

자동화 = 판단 제거 × 수작업 제거 ○

기존 ERP · EAM

기록 ○ 근거 생성 ×

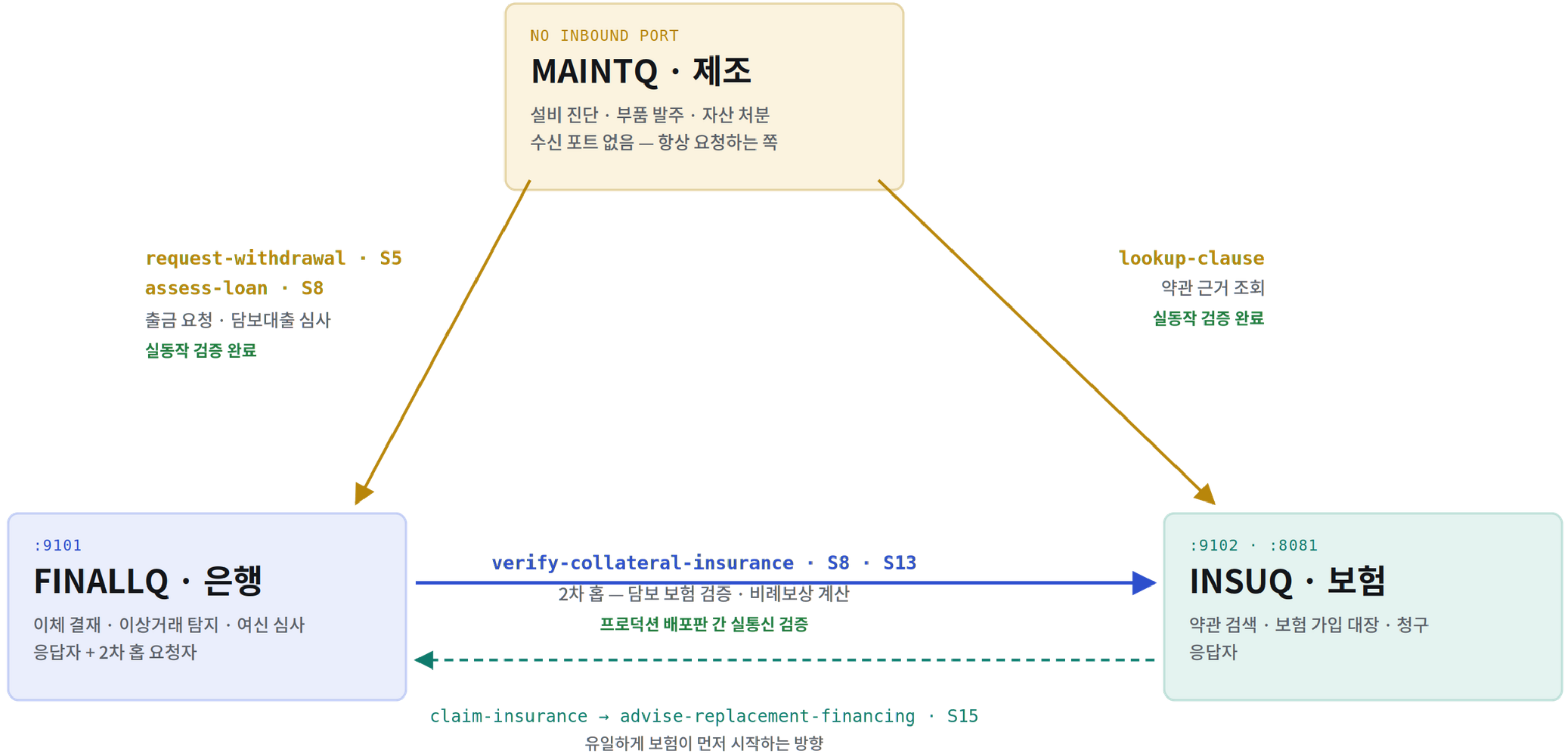
범용 AI 챗봇

응답 ○ 권한·감사·승인 ×

QMesh

근거 조립 + 승인 계단 + 감사 기록

외부 노출 = A2A 포트 3개



시스템 별 AI 구조 비교



셋 다 "AI 에이전트"라고 부르지만 구조가 전부 다릅니다

필요한 곳에만 LLM을 씁니다 — 금융의 도구 오케스트레이션은 규칙으로 충분했고, 규칙이 더 안전합니다

전체 아키텍처 노드

MaintQ · 설비보전NO INBOUND PORT

FRONTEND

Next.js · 화면 16라우트 (진단 콘솔 · 승인 큐 · A2A 이력)

BACKEND

FastAPI routers/chat.py · routers/po.py · routers/a2a.py

agent/loop.py run_turn() — 단일 에이전트 루프

agent/llm.py Gemini 2.5 Flash

MCP · 별도 프로세스

mcp_server FastMCP — core 7 / 확장 11 = 18 도구

쓰기 3종만 draft INSERT · DB 트리거로 차단

A2A OUT · 발신

a2a/client.py call_skill() + auth_header.py

DATA

PostgreSQL (pgvector) · 24테이블 · 매뉴얼 1,229청크

FinAllQ · 은행 · 증권:9101

FRONTEND

React 19 · 화면 33개

A2A IN · 수신

a2a_adapter 스킬 7종 · insuq_client.py (2차 홈 발신)

ORCHESTRATOR

mcp/hub LangGraph StateGraph — 노드 6개

plan · tool_select · tool_call · observe · synthesize · respond

⚠ 이 그래프 안에 LLM 없음 — 규칙·템플릿 기반

AGENT NODES

fds.score · portfolio.rebalance · sms.classify

BACKEND

backend-core (Spring, :8082) · 60+ REST · RBAC · TOTP

DATA

PostgreSQL / pgvector · Redis 캐시

InsuQ · 보험약관 RAG:8081 · :9102

FRONTEND

Next.js · 설계사 고객관리 화면

A2A IN · 창구 2개

backend A2aController (:8081) — 스킬 5종

a2a_adapter (:9102) — lookup-clause

BACKEND

Spring (:8081) 정책 원장 · Task 상태머신 · 감사로그

AI ENGINE

FastAPI /qa · /qa/stream

agent/graph.py LangGraph — 노드 2개 (route · clarify)

pipeline.py 검색 → 방어선① → 생성 → 방어선② → 판정

DATA

Qdrant Cloud · 실손 + 화재재물 6,505포인트

A2A 경계 — HTTP + Agent Card

밖으로 나가는 것은 A2A 창구뿐 — 화면·백엔드·DB는 경계를 넘지 않는다

세 스택이 전부 다릅니다. 벡터 스토어만 봐도 Qdrant / pgvector로 갈립니다 — '별개 회사'라는 전제를 코드로 지켰습니다

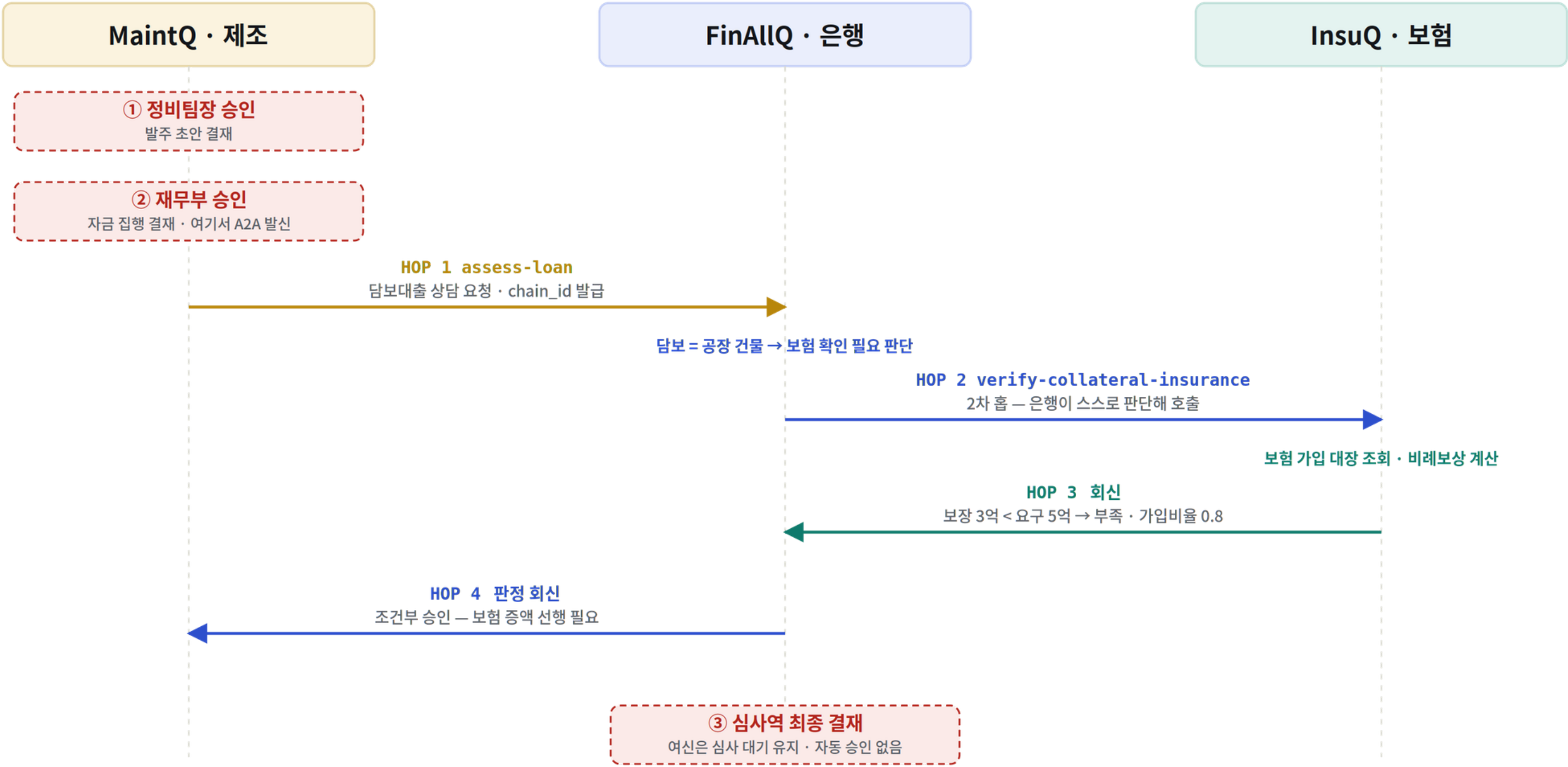
MaintQ는 수신 포트가 없습니다 — 스킬을 노출하지 않는 항상 요청자입니다

시나리오 목록

구분(번호)	시나리오	경로	성격	상태
S1—S4	진단 → 재고 → 견적 → 발주 초안 → 승인 / 대체품 / 반복고장 / 모른다	MaintQ 내부	도구 오케스트레이션	실동작 검증
S9·S10·S29	처분 법정조건 차단 · 근거 번들 서명 · 수리 증빙 서명	MaintQ 내부	확정 방어	실동작 검증
S19—S23	기업 온보딩 · 이체 결재 · 이상거래 탐지 · AI 비서 · 통합자산	FinAllQ 내부	결재 · 통제	구현됨
S24—S28	근거 인용 · 되묻기 · 확인 불가 거부 · 조항 멀티홉 · 화재재물 트랙	InsuQ 내부	RAG · 판정	구현됨
S5	발주 확정 → 출금 요청 (2단 승인)	Maint → Fin	요청 + 승인	실동작 검증
S7	고장이력 연계 화재보험 갱신 상담	Maint → Insu	상담 + 데이터연계	실동작 검증
S11	설비 처분 → 보험 목적물 변경 통지	Maint → Insu	통지 + 정산	구현됨
S15	화재 멸실 → 보험금 청구 → 승인 큐	Insu → (Fin)	청구 + 사람승인	실동작 검증
S8	담보 대출 심사 시 보험 검증 (멀티홉 릴레이)	Maint → Fin → Insu	3사 릴레이	▶ 시연
S6·S12·S13·S14·S16	환헤지 상담 · 매각대금 정산 · 중고설비 담보심사 · 위험등급 통지 · 조달 비교	경계 넘음	—	수신부 완료 · MaintQ 발신 트리거 대기

시나리오 흐름

설비 이상이 감지된 순간부터 은행이 조건부 승인을 회신하기까지, 사람이 세 번 개입합니다.



결재 문서 목록

에이전트가 만들고 사람이 결재하는 문서.

생성 → 결재 → 회신

단계	문서	생성 · 수신	서명 주체	핵심 필드
01	설비 이상 진단 보고서	MaintQ 생성	정비사 작성 · 팀장 검토	근거 인용 · 안전경고
03	정비 · 부품 발주 요청서	MaintQ 생성	정비사 기안 · 팀장 승인	재고 · 대체품 · 견적
04	자금 집행 요청서	MaintQ → FinAllQ	3단 결재 · 재무부 승인	FDS · 자기결재 확인
08	담보대출 심사 회신서	FinAllQ	심사역 최종 결재	보장한도 · 부보비율 · 처리 이력

문서 신뢰성

결재 단위 = 문서 · 감사 시 서명 대상 특정 · 채팅 로그 불가

승인 기록이 남아야 하고, 나중에 감사에서 "무엇에 서명했는가"가 특정되어야 합니다.
받지도 않은 값을 요구할 때, “확인되지 않음”으로 채웁니다.

서명 시점 해시 재대조

근거 해시 재검증 · 근거 변경 후 서명 차단

서명하는 순간, 문서에 실린 근거의 해시를 다시 대조합니다.
근거가 바뀐 뒤 서명되는 상황을 구조적으로 막습니다.

PART 04

MaintQ · 설비보전 에이전트

진단 · 근거 조립 · 초안 생성 확정은 사람

설비보전 에이전트 기능

: 근거 제시 · 사람 확정

진단

에러코드 × 매뉴얼 이원화

정의 = 정확 일치 . 점검 절차는 매뉴얼 RAG.

확실/추정 구분 표시

매뉴얼 검색은 키워드+dense 하이브리드, 1,229청크 · 3기종(iG5A·S100·IE5)

조달

재고 → 대체품 → 견적

부품 특정 · 재고 조회

재고가 0이면 대체품을 탐색하고, 공급사별 견적을 비교해 발주 초안을 만듭니다.

안전

위험 작업 경고 자동 삽입

위험 언급 시 경고 삽입 · 근거 부재 시 발행 거부

이력

반복 고장 감지

동일 에러가 30일 안에 3회 발생 시,
근본 원인 점검 모드로 전환, 발주 보류

자산

수리 / 교체 / 매각 판단

보전 지표 기반 자산 판단

처분 시 법정 조건 미충족을 차단 및 해결 방법을 안내

한계

모르는 건 모른다고 한다(확인 불가)

미등록 에러코드에 유사 코드 추측 금지
환각률 0.0%

16

도구 (코어 9 + 확장 9)

24

DB 테이블 + seed

18

화면 라우트

0.0%

환각률

자산 생애주기

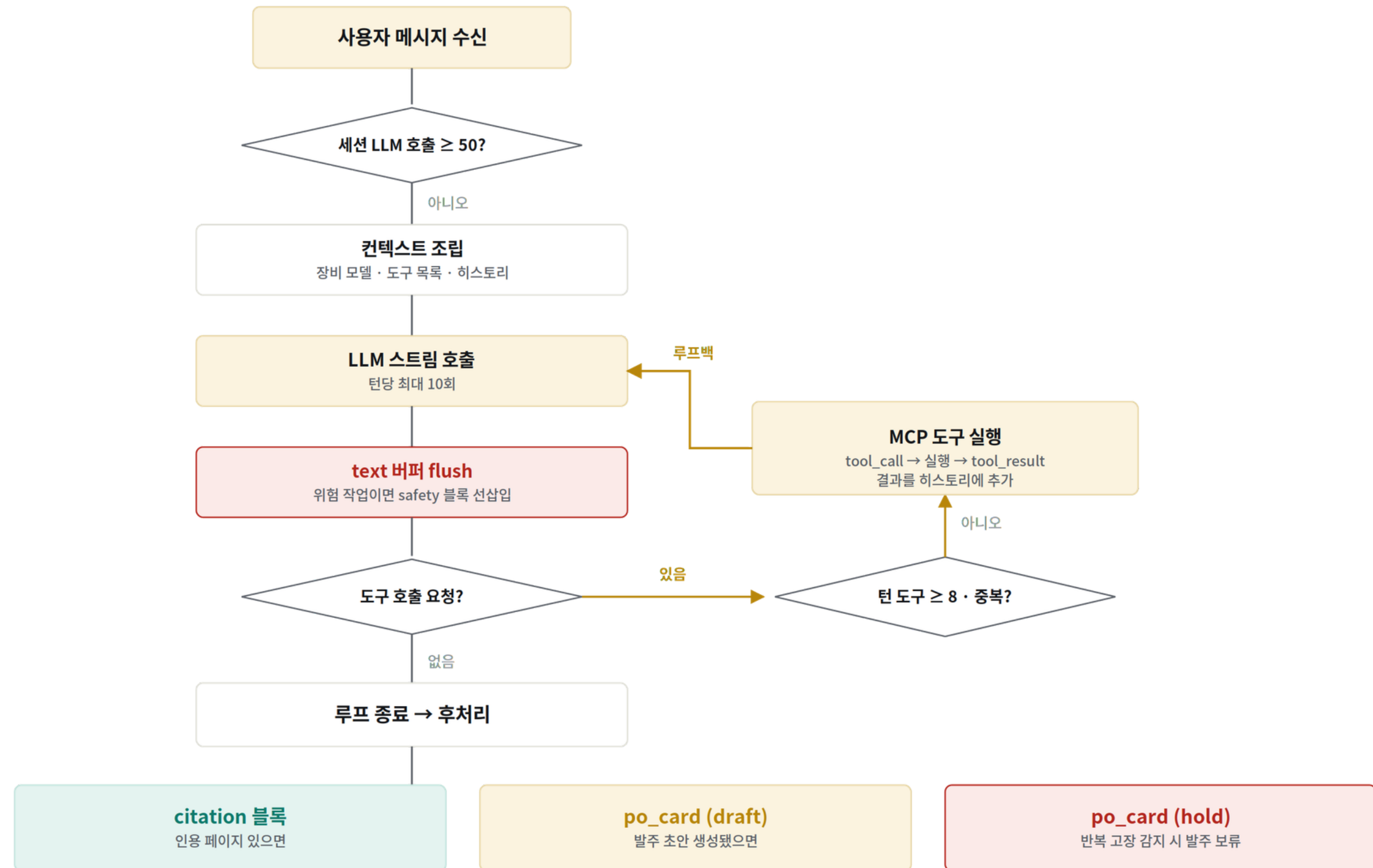
MaintQ 자산 생애주기 — 진단에서 감시까지



에이전트는 ①~④를 채우고, 사람이 ⑤에 서명하고, ⑥~⑧이 기록으로 남습니다

A2A는 ⑤ 이후에만 나갑니다 — 승인 없이 경계를 넘는 경로는 없습니다

Agent Loop



근거 3계층 — 사실 · 해석 · 서명

조문은 사실이고, "이 사안에 이렇게 적용된다"는 해석입니다. 해석은 틀릴 수 있어서 물리적으로 분리했습니다

계층 1 · 사실

이건 바뀌지 않는다

법령 스냅샷

- 법제처 API로 수집한 조문 원문
- 파일(git) 보관 · append-only
- text_hash로 잠금



계층 2 · 해석

이건 틀릴 수 있다 — 그래서 버전이 있다

해석 룰 카탈로그

- "이 조건이면 처분 불가" 같은 판정 규칙
- rule_version으로 개정 이력 관리
- 근거 없는 룰은 로드 단계에서 거부



계층 3 · 서명

이건 사람이 책임진다

사람의 결정

- 서명 시점에 근거 번들 해시 재대조
- 판정과 다른 결정도 허용 — 단 사유 필수
- 불일치하면 409 evidence_changed

"조문은 사실이고, 적용은 판단이다" — 섞어두면 나중에 무엇이 사실이었는지 구분할 수 없습니다

계층 1은 파일과 git으로, 계층 2는 버전으로, 계층 3은 해시로 잠급니다

근거가 바뀐 뒤 서명되는 상황을 구조적으로 막습니다

근거 번들 5키

서명 대상이 되는 문서 구조

laws[]

인용된 조문 + text_hash

rules[]

발화한 룰 + rule_hash

evaluated[]

평가한 전 룰의 ID · 판정

contracts[]

계약 근거 (해시 미고정 명시)

facts

판정에 쓰인 사실값

자동화 경계선

LAYER 1 · 읽기

조회 · 검색 도구

- 에러코드 록업
- 매뉴얼 RAG
- 재고 조회
- 견적 조회
- 이력 조회
- 법령 조회

완전 자동

LAYER 2 · 판단

계산 · 판정 도구

- 반복고장 판정
- 보전지표 산출
- 수리/교체/매각
- 법정조건 검사
- 안전경고 발행

자동 — 근거 없으면 발행 거부

LAYER 3 · 쓰기

초안 생성 도구 — 3개뿐

- 발주 초안
- 처분 초안
- 수리 초안

draft-only — 사람이 확정

사람 확정 — 승인 · 서명

- 권한 (403)
- 스키마
- 순서
- DDL CHECK

확정 방어 4층

MaintQ 트러블슈팅

문제	원인 · 판단	조치	결과
평문 자격증명이 커밋됨	외부 응답을 저장하며 요청 헤더까지 함께 남김	커밋 10개 히스토리 재작성 · 응답 본문만 allowlist로 저장하는 구조로 변경	구조적 방지
같은 질문에 답이 흔들림	도구 루프의 비결정성. temperature 0인데도 실행마다 갈림	제거 대신 측정으로 전환 3회 반복 후 「안정실패 → 안정통과」 칸만 개선 인정	조사 계속
매뉴얼 결측 34건	PDF에서 조치 항목이 추출되지 않음	2차 판독기(Elice AI)로 34페이지 재판독 (1,530원)	확정 24 · 회수 0
평가가 너무 느림	20문항 반복에 132초 - 개선 사이클이 돌지 않음	LLM 응답 카세트(캐시). 단 재생 턴은 분모에서 제외, 기본 모드는 캐시 차단	132초 → 61초
부품 품번 특정 40%	제조사가 소모 교체품 품번을 공개하지 않음	자동화 불가로 판정 - 사람이 제조사에 문의하는 경로로 남김	해결 불가 · 명시

PART 05

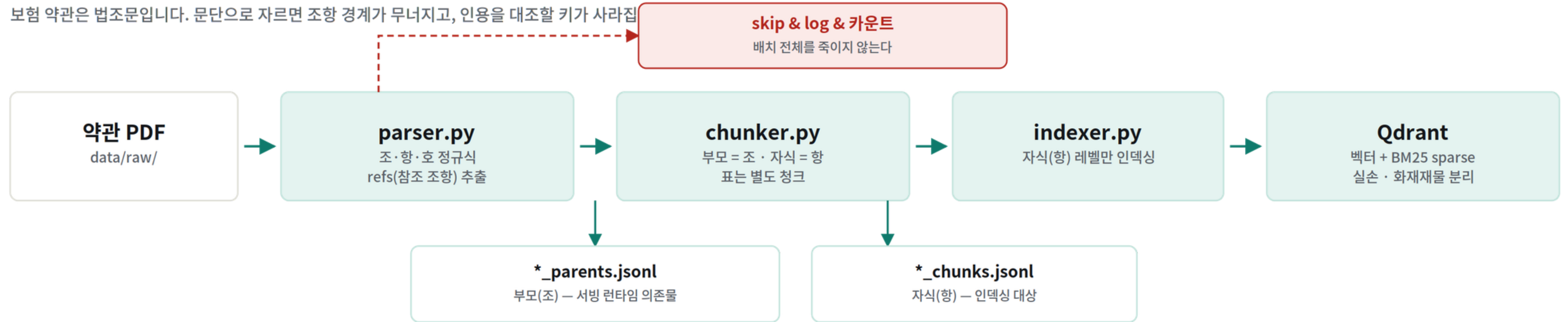
InsuQ · 보험약관 RAG 에이전트

거부할 줄 아는 에이전트

약관 인제스트

약관 인제스트 — 문단이 아니라 조문 구조로 자릅니다

보험 약관은 법조문입니다. 문단으로 자르면 조항 경계가 무너지고, 인용을 대조할 키가 사라집니다



자식으로 찾고, 부모로 답한다

검색·리랭킹은 전부 자식(항) 레벨입니다.
부모(조) 전체를 붙이는 건 생성 직전 한 번뿐입니다.

실손과 화재재물은 인덱스가 다르다

약관 파트가 다르면 조 번호가 겹칩니다.
섞으면 다른 파트 조항을 지어내도 검사를 통과합니다.

인용 대조 키

(policy_part, article_no) 가 일치해야 답이 나옵니다.
파트를 빼면 환각 탐지에 구멍이 생깁니다.

구조를 지켜야 인용을 대조할 수 있고, 대조할 수 있어야 환각을 잡을 수 있습니다

거부할 줄 아는 RAG

근거 없으면 확인 불가

파싱

조 · 항 · 호 구조 인식

문단 분할 × · 법조문 구조 분할 O

약관 PDF를 법조문 구조로 분할

자식(항)으로 검색하고 부모(조)로 생성

인용

근거 조항 필수

전 답변 인용 첨부

대조 키 불일치 시 유보」로 강등

거부

2중 방어선

LLM 호출 전 차단 + 생성 후 검증

근거 0건이면 즉시 거부하고, 생성 후에 인용을 다시 검증. 거부 정확도 100%.

되묻기

슬롯 부족 시 질의

상품 · 특약 · 가입시기

A2A의 input-required 연동

판정

3단계 - 높음 · 낮음 · 유보

지금 가능성을 세 값으로만 답함.

그 밖의 표현은 코드상 존재하지 않습니다.

격리

타 고객 계약 비인용

상품 필터를 5계층에 배선, 캐시 키에도 포함

캐시로 안전장치가 우회되지 않게 하기 위해서임

96

골든셋 문항 (사람 검수)

52

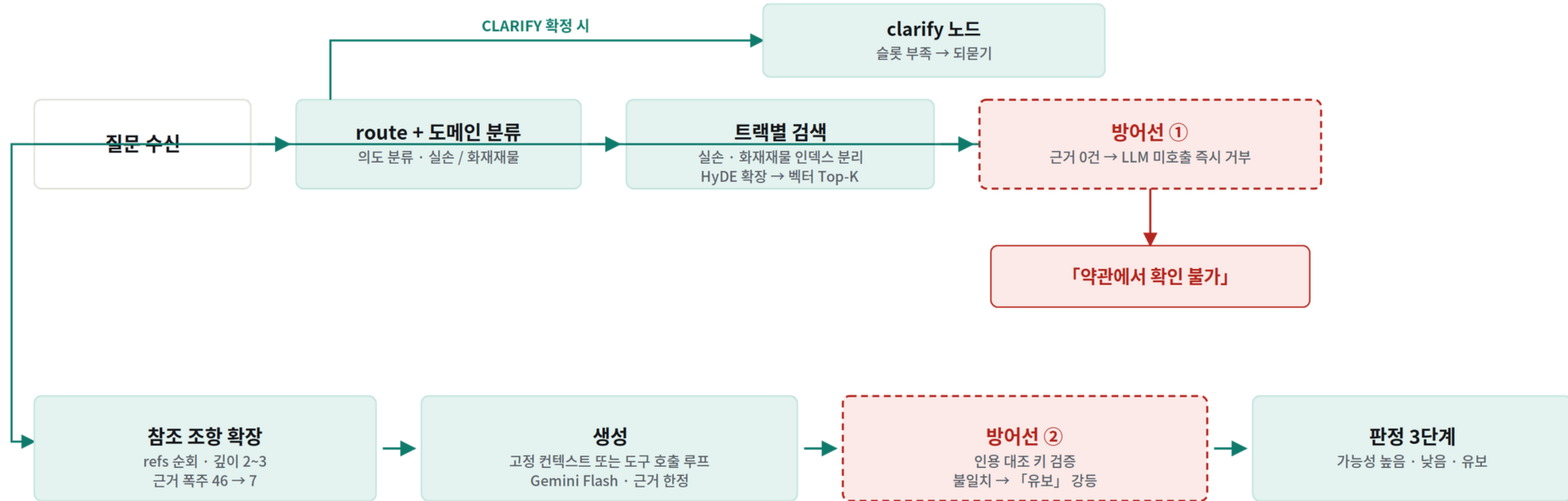
기록된 실험 (EXP)

100%

거부 정확도 · 오탐 0%

검색 → 생성 → 거부 파이프라인

라우팅 → HyDE 확장 → 벡터 검색 → 방어선 → 생성 → 재검증



인용 대조 키

(policy_part, article_no) 가 일치해야 통과
약관 파트를 빼면 다른 파트의 같은 조 번호를 지어내도 검사를 통과한다

도메인이 애매하면

두 트랙 각각 검색한 뒤 more_conservative()로 병합합니다
— 둘 중 보수적인 쪽을 택합니다. 근거에 도메인을 태깅합니다

실손과 화재재물은 인덱스 · 골든셋 · 그래프가 전부 분리된 별도 트랙입니다

거부 정확도 100% · 오탐 0%

성능 평가



6종을 같은 골든셋에서 재고, 1종만 채택했습니다

기각 사유가 각각 다릅니다 — 품질 미달 · 시간 초과 · 효과 없음

하이브리드는 기준선보다 나빠졌고, 리랭커는 품질은 올랐지만 30초 예산의 3배였습니다

트러블슈팅 : 지표 분해 · 원인 분리

문제	파고들어 보니	조치	결과
응답이 44.7초	타임아웃 25초 → 재시도 백오프 7초 추가 → 다시 초과 → 재시도 악순환	구조로 해결 불가. 모델 교체	44.7s → 10.4s
빈 답변이 정답으로 집계	추론 토큰이 max_tokens를 다 먹고 본문이 비었는데 정상 처리됨	전용 예외 신설 · 토큰 상향 · 회귀 테스트	측정 신뢰도 회복
과잉 거부 4건	원인이 세 갈래로 섞여 있었음 검색 실패 / 생성 오류 / 간헐적 흔들림. 해결책이 각각 다른데 한 숫자였음	정답 조항이 검색됐는지로 지표를 분리	원인별 분류 가능
정상 인용이 환각 판정	약관명에 괄호가 붙어 문자열이 안 맞음. 코퍼스의 16.1%(356 / 2,214 청크) 영향	느슨하게 풀지 않고 「거부」 대신 「유보」로 강등	탐지 구멍 없이 해결
과잉 거부가 또 남음	보이지 않는 추론 토큰이 예산을 소진 - 가시 토큰 600인데 length로 종료	토큰 예산 3000 → 6000 (실서빙에도 전파)	0.1154 → 0.0577

기법 추가로 효과를 봤더라도, 다른 곳에 근거 없이 적용하지 않았습니다.

토큰 상향이 화재 트랙에서 효과를 봤지만, 실손 트랙은 과잉 거부가 애초에 0건이라 개선 여지가 없다는 걸 측정으로 확인하고 변경하지 않았습니다.

PART 06

FinAllQ · 은행 · 증권 에이전트

결재 라인 · 이상거래 탐지 · 직무 분리 기구축

이체 흐름

FinAllQ 이체 — 사람은 한 곳에서만 멈춥니다

요청은 A2A로 들어오고, 판정·한도·탐지·실행은 자동입니다. 사람이 하는 건 승인 하나뿐입니다.



사람은 한 곳에서만 멈춘다

결재자 판정 · 한도 검사 · 이상거래 탐지 · 실행은 전부 자동입니다. 사람이 하는 건 승인 하나뿐입니다. 자동화를 덜 한 게 아니라, 멈출 지점을 고른 것입니다.

2차 인증은 대신 못 낸다

A2A로 인증 코드를 대신 제출하는 기능은 의도적으로 만들지 않았습니다. 서비스 계정이 대신 내면 2단계 인증이 1단계로 무너집니다.

요청이 에이전트에서 왔다면 에이전트 동작이다

내부 처리가 REST 호출이더라도, 이 흐름을 시작한 것은 MaintQ 에이전트의 판단입니다. 사람이 화면을 연 게 아니라 에이전트가 스스로 필요를 판단해 요청을 만들었습니다.

자동 4단계 · 사람 1지점

— 자동 승인 경로는 코드 어디에도 없습니다

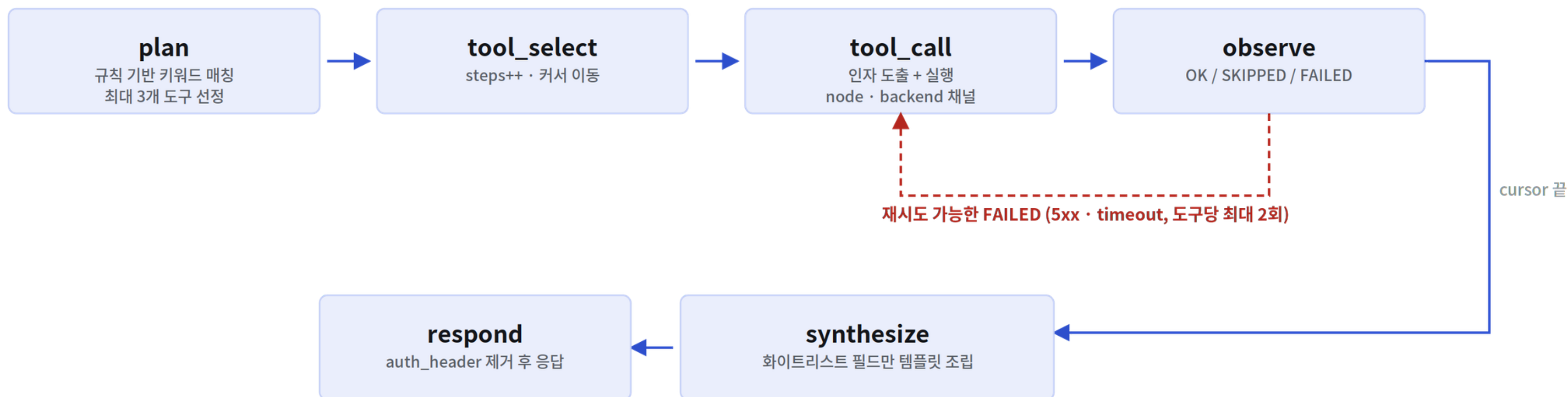
이상거래 탐지는 승인 뒤에 듭니다. A2A로 들어온 요청도 이 순서를 건너뛰지 않습니다.

A2A 적용 6개의 통제 장치

기존 통제 → A2A 확장

장치	내용	A2A에서의 역할
직무 분리 매트릭스	판정 축을 곱한 18칸 중 허용은 4칸. 자기결재 금지, 관리자는 결재자가 아님	외부 에이전트 요청도 같은 매트릭스를 통과해야 합니다
이중 방어	규칙(한도) → 모델(이상거래 탐지). 순서 자체가 통제	A2A 요청이 모델을 우회할 수 없습니다
화이트리스트 차단	감사 기록과 AI 페이로드는 등재된 키만 통과	A2A 페이로드에도 그대로 적용됩니다
append-only 감사	성공뿐 아니라 차단·거부 시도까지 기록. 롤백과 무관하게 보존	추적 ID만 없으면 멀티홉 추적이 됩니다
차단기 (fail-soft)	AI 장애가 코어로 번지지 않음	남의 에이전트가 죽어도 내 코어는 살아야 합니다
테넌트 격리	다른 소속 자원은 "권한 없음"이 아니라 "안 보임"	요청자 회사 밖 자원이 새지 않습니다

FinAllQ LangGraph



이 그래프 안에는 LLM이 없습니다

plan(어떤 도구를 부를지)도, synthesize(응답 조립)도 전부 규칙과 템플릿으로 결정됩니다 — 필요한 곳에만 LLM을 씁니다

왜 규칙인가

돈이 걸린 도구 선택이 실행마다 흔들리면 감사 자체가 성립하지 않습니다. 같은 입력에는 같은 도구가 같은 순서로 불립니다.

FinAllQ — 세 시스템 중 유일한 실제 LangGraph 오케스트레이터 (노드 6개)

하위 에이전트 노드 3종: fds.score · portfolio.rebalance · sms.classify — 전부 결정론적 서비스

FinAllQ 트러블슈팅

같은 문제가 두 번 생겼습니다

"도구는 만들었는데 화면이 없는" 상태를 없앴습니다(도구 20종 중 3종 → 0종).

그런데 다음 스프린트가 여신 심사 API 4개를 만들면서 화면을 만들지 않아 같은 문제가 자리를 옮겨 재발했습니다.

발표 문서에도 그대로 적었습니다 — "지금 여신은 curl로만 쓸 수 있다"

어떻게 달았나

여신 화면을 탭 구조로 묶어 시뮬레이터 → 신청 → 내 신청 → 심사가 한 흐름으로 이어지게 했습니다.

그리고 라우팅 테스트가 앱 전체를 렌더해서 탭을 실제로 클릭하게 바꿨습니다.

이전에 페이지 컴포넌트가 어느 테스트에도 import되지 않아, 완주가 검증되지 않은 채 통과한 사고가 있었기 때문입니다.

문제	원인	조치 · 결과
도구는 실행되는데 답변에 반영 안 됨	도구 결과를 답변으로 옮기는 경로 자체가 명세에 없었음	화이트리스트 기본 거부 + 실데이터 참조. 민감정보 노출 0건
목록 조회가 느림	N+1 쿼리	응답 시간 68% 단축
페이지 경계에서 항목이 새거나 중복	정렬 키가 하나라 동률에서 순서가 흔들림	이중 키 정렬 — 버그 8건 해결
동시 승인 시 갱신 유실	낙관적 잠금 부재	비교 후 교체 방식 도입 — 유실 0건

PART 07

프로젝트 간 통신 · 실행 결과

경계 통과 전 계약 · 데이터 · 추적

A2A : 계약 기반 위임

공개 = 능력 / 비공개 = 구조

01

Agent Card

agent card에 수행 가능 작업만 공개
내부 구조 비공개

02

Task 상태

submitted → working → completed
중간 input-required(사람 승인 대기)

03

추적 ID

request_chain_id가 최초 요청~마
지막 회신까지 추적
성공·실패·타임아웃 전부 기록

04

중복 방지 키

같은 요청이 두 번 들어와도 두 번 실행되지 않습니다. 3중 복합키로 판정합니다.

신원은 두 층입니다

Bearer 토큰 + X-A2A-Partner-Id — actor 증명 (= 인증)
payload requester 식별자 — subject 지정 (= 대상)

식별자 보유 ≠ 인증
이 구분을 흐리면 남의 회사 자원을 조회 가능해짐

자금·계약 스킬은 즉시 완료 불가

보험금 청구 스킬의 requires_human_approval은 고정 값(계산 ×)
조건에 따라 false가 될 경로가 없습니다.

승인 대기 상태로 넘어가 사람의 결재함에 들어갑니다.

감사 기록 = 해시 다이제스트만 · 원문 미저장

계약3계층 + 연결승인

계약은 3계층

무엇을 약속했는가

① Agent Card

docs/agent_cards/*.json

이 시스템이 노출하는 스킬 목록 · 인증 방식 선언

② JSON Schema

docs/schemas/{skill-id}.json

스킬별 요청 · 응답 필드 · 타입 · 필수 여부

③ 어댑터 구현

backend / a2a_adapter (each repo)

Pydantic 검증 후 실제 백엔드로 연결

계약 원본(SSOT)은 A2A_Q 한 곳에만 둡니다

여러 레포에 복제하면 한쪽만 고치고 잇는 drift가 생깁니다

스킬 호출 전, 사람이 먼저 승인합니다

어떤 스킬도 이 4단계 전에는 발생할 수 없다

① 등록

사람

은행 관리자가 공장을 기업고객으로 등록

② 연결 승인

사람

"이 회사와 통신을 허용한다"

③ 자격증명 발급

사람 → 기계 경계

허용 작업 범위 · 한도 · 유효기간 함께 발급

④ 스킬 호출 시작

기계

에이전트가 토큰을 받아 요청 전송

①②는 무겁고 1회성 · ③④는 그 뒤 반복되는 가벼운 절차

자격증명은 회사가 아니라 회사를 대표하는 담당자(사람)에게 발급됩니다

멀티홉 실행

사람 결재 3회 · 에이전트 4홉 · 추적 ID 1개

홉	주체	스킬 · 동작	결과	Task 상태
—	사람	정비팀장 발주 승인	초안 → 승인됨	사람 결재 ①
—	사람	재무부 자금 집행 승인	결재 완료	사람 결재 ②
HOP 1	MaintQ	상담 요청 조립 · 위임	요청 전송	submitted
HOP 2	FinAllQ	assess-loan	담보 = 공장 건물 → 보험 확인 필요 판단	working
HOP 3	InsuQ	verify-collateral-insurance	보장 3억 · 요구 5억 → 부족 (부보비율0.8 · 비례보상 반영)	completed
HOP 4	FinAllQ	결과 반영 · 판정	조건부 승인 : 보험 증액 선행 필요	completed
	MaintQ	회신 수신 · 표시	"대출 승인, 단 보험 3억 → 5억 증액 필요"	completed
	사람	심사역 최종 결재	여신 상태는 심사 대기 유지. 자동 승인 없음	사람 결재 ③

추적

4홉 동일 chain_id
시작 주체·경로 기록

분기

보장 < 대출 → 거절 ×
조건부 승인 0
다음 조치 명시

연결

조건부 승인 → 보험 증액 → 대출 집행 · 삼각형 폐합
이 조건부 승인은 보험 증액 상담으로,
증액이 끝나면 대출금 집행으로 이어집니다.

A2A 스킬 현황

MaintQ는 항상 요청을 시작하는 쪽이라 A2A 수신 포트 자체가 없어 스킬을 노출하지 않습니다.

제공 측	스킬	시나리오	성격	상태
FinAllQ	request-withdrawal	S5	요청 접수 (실행 X)	종단 검증
FinAllQ	assess-loan	S8	사전 심사 · 2차 홉 트리거	실동작
FinAllQ	상담 · 정산 · 중고설비 등 5종	S6 · S12 · S13 · S15 · S16	—	구현 완료
InsuQ	verify-collateral-insurance	S8 · S13	순수 조회 + 비례보상 계산	구현 완료
InsuQ	advise-policy-renewal	S7	상담 + 고장이력 연계	구현 완료
InsuQ	notify-asset-change	S11	목적물 변경 통지	구현 완료
InsuQ	claim-insurance	S15	보험금 청구 + 사람 승인	E2E 검증
InsuQ	notify-risk-change	S14	위험등급 변동	구현 완료
InsuQ	lookup-clause	범용	약관 근거 조회	구현 완료

보험 쪽 병목은 "창구"가 아니라 "창고"였습니다

스킬 5종 중 4종이 정책 원장(누가 어떤 보험에 가입했는가)을 조회합니다. 원장이 없으면 창구를 열어도 빈 응답만 나옵니다. 그래서 원장부터 만들었습니다.

연결해보니 버그가 나왔습니다

각 레포에서 개별 테스트는 전부 통과했는데, 실제로 세 개를 붙이자 3건이 깨졌습니다 : InsuQ 헤더 누락, requester 누락, assess-loan 계약 불일치. 개별 검증으로는 안 잡히는 종류였습니다.

PART 08

성과 · 아쉬운 점

무엇을 증명했고, 무엇을 놓쳤는가.

증명한 것 네 가지

01

경계를 넘어 요청이 오가고, 근거와 함께 회신된다

DB 직접 접근 0 · 3사 요청 완주 · 약관 조항 인용 첨부
서로의 DB를 한 번도 직접 보지 않은 채, 세 회사가 하나의 요청 흐름을 완주했습니다. 회신에는 약관 조항 인용이 붙습니다.
verify-collateral-insurance·notify-risk-change·claim-insurance 세 스킬 검증 완료

02

연결해도 통제를 잃지 않는다

직무분리 · 이중방어 · 감사기록 A2A 동일 적용 · 인증 대행 경로 부재
직무 분리, 이중 방어, 감사 기록이 A2A 요청에도 그대로 적용됩니다. 2차 인증을 대신 내주는 우회 경로를 만들지 않았습니다.

03

모른다고 말할 줄 안다 (거부 능력 확보)

거부 정확도 100% · 오탐 0% · 환각률 0.0%
거부 정확도 100%, 오탐 0%, 환각률 0.0%. 남의 에이전트에게 그럴듯한 오답을 주지 않는 것이 A2A에서 가장 중요한 안전장치입니다.

04

모든 기술 선택을 측정으로 검증했다 (측정 기반 선택)

기법 6종 비교 · 1종 채택 · 기각 사유 3분류 기록
검색 기법 6종을 같은 골든셋에서 비교해 1종만 채택했습니다. 기각 사유를 품질 미달·시간 초과·무효과로 구분해서 기록했습니다.

12

구현된 A2A 스킬 (FinAllQ 7 · InsuQ 5)

3

MaintQ 발신 실동작 스킬

0

자동 승인 경로

3,600+

자동 테스트 · 회귀 0

한계

세 도메인 병행 → 한 사람의 하루 미추적

세 도메인을 동시에 파느라, 한 사람의 하루를 깊게 보지 못했습니다

기능은 각각 돌아가는데, 정비팀장 한 명이 아침에 출근해서 이 시스템으로 무엇을 하는가를 끝까지 따라가 본 적이 없습니다. 각 프로젝트의 기능 구현에 급급해, 사용자 맞춤형 UI나 워크플로우인지 고민할 시간이 부족했다는게 아쉽습니다.

남은 것 01

외부 응답 표시 화면

A2A 회신이 추적 테이블과 콘솔에만 남습니다.
렌더링 화면이 없습니다.

남은 것 02

머신 신원 · 파트너 인증

설계는 2층으로 문서화했지만,
실제 인증은 아직 목업 토큰 단계입니다.

남은 것 03

검색 목표 미달

Hit@5가 목표 0.85에 골든셋 1문항 차이로 도달

남은 것 04

스킬 5종 스텝

상담·정산·중고설비 계열이 아직 열리지 않았습니다.
위험등급 통지는 계획에서 누락됐습니다.

남은 것 05

실행 계층 미완

여신은 승인까지고 실제 자금 이동은 별도입니다.
계좌에 여신·수신 구분이 없습니다.

남은 것 06

도구 루프 비결정성

같은 질문이 실행마다 갈리는 현상.
원인이 외부라 제거하지 못하고 측정으로 관리 중입니다.

계획

처음에는 "위험이 낮은 것부터"로 정했는데, 실제로 재보니 "준비된 것부터"가 맞았습니다.

처음 세운 순서 — 위험도 기준

① 조회·상담 (돈 안 움직임) → ② 출금 (2단 승인)

착수하자마자 막혔습니다. 조회가 안전한 건 맞지만, 조회할 대장 자체가 없었습니다.

실측 후 순서 — 준비도 기준

① 선행조건 없는 스킬 → ② 레일이 이미 깔린 것 → ③ 창고를 만들어야 하는 것

"안전한 것"과 "준비된 것"은 다릅니다.

우선	할 것	왜 이 순서인가	막고 있는 것
1	MaintQ 발신 트리거 9종	상대는 다 준비 · 이것만 열면 시나리오 9개가 한 번에 열림	없음 — payload 빌더 패턴 3종 이미 있음
2	lookup-clause 창구 일원화	임시/정식 이중화 해소	InsuQ H09 결정 필요
3	인증 헤더 실전 검증	Bearer 스킴이 아직 안 통과해봄	정식 수신부 열려야 함
4	하이브리드 가중치 확정	임시값 1.0/1.0	eval 실측 필요
5	InsuQ 수신함 UI	backend 완료	-

상대는 준비를 마쳤습니다. 남은 건 우리가 보내는 일입니다

자동화 : 판단은 남기고, 수작업만 제거

에이전트는 근거를 모으고, 초안을 만들고, 경계를 넘어 물어봅니다.

에이전트의 추론을 보고 판단, 승인하는 것은 사람입니다. 그 경계를 권한과 스키마와 제약으로 강제했습니다.

증명한 것

경계를 넘어 근거와 함께 회신된다

지킨 것

자동 승인 경로 0건

남은 것

사람에게 닿는 마지막 화면

APPENDIX

부록

질의응답에서 꺼낼 자료. 발표 중에는 넘기고, 질문이 나오면 찾아갑니다.

3사 아키텍처 · ERD · 다이어그램

MaintQ

① 아키텍처 · ERD ② 구조 자료 ③ 다이어그램 목록
에이전트 도구 계층도 (읽기 / 판단 / 쓰기)
도구 호출 그래프 · 실행 추적 타임라인
24개 테이블 ERD
확정 방어 4층 도식
근거 3계층 (법령 / 해석률 / 서명)

FinAllQ

이체 상태 전이도
직무 분리 매트릭스 (18칸 중 허용 4칸)
여신 도메인 ERD + 이체 연결
도구 오케스트레이션 그래프 (20종)
이상거래 탐지 이중 방어 흐름

InsuQ

3층 구조 (프론트 → 백엔드 → 엔진)
약관 청킹 트리 (조 → 항 → 호)
라우터 분기도 · 상태 그래프
2중 거부 게이트 흐름
검색 실험 비교 그래프 · 산점도

A2A 공통

① 지표 요약 ② 측정 결과 전량 ③ 지표 · 목표 · 달성

삼각형 통신 다이어그램 · Task 생명주기 상태도 · 신원 2층 구조도 · 멀티홉 시퀀스
다이어그램 · 추적 ID 전파 경로

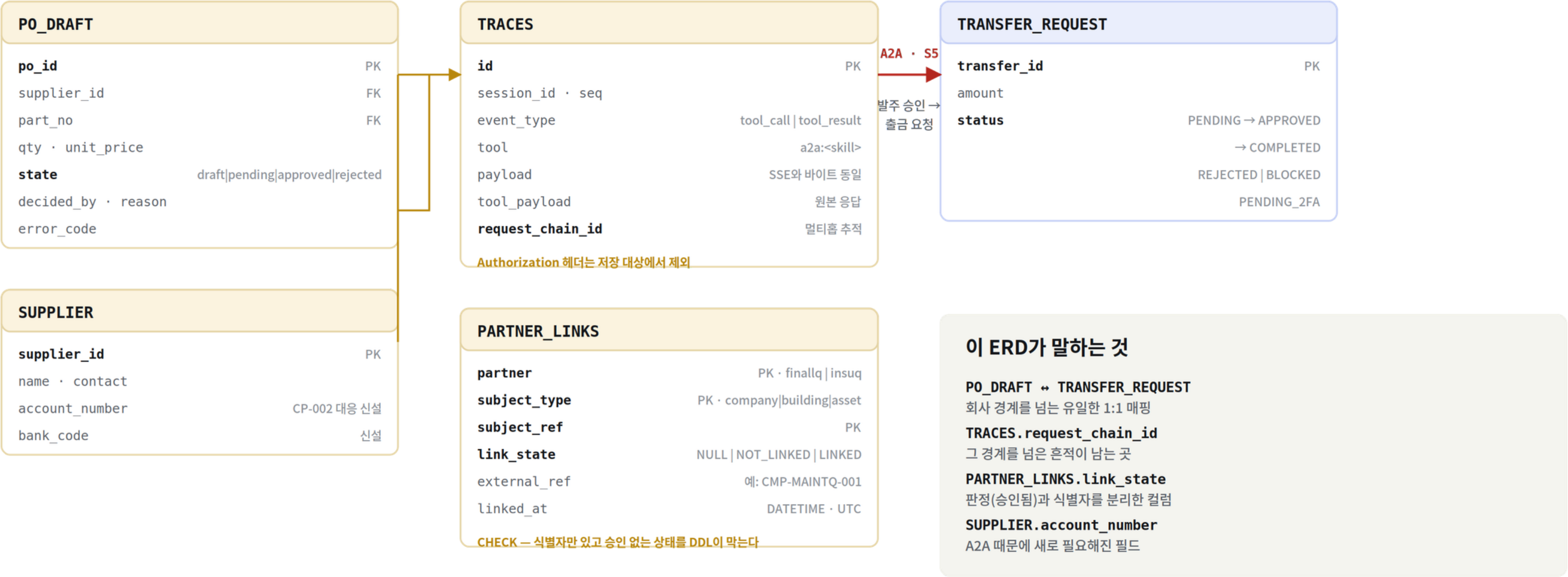
산출 문서 실물

설비 이상 진단 보고서 · 정비 부품 발주 요청서 · 자금 집행 요청서 · 담보대출 심사
회신서 (각 서식 PDF)

A2A가 건드리는 테이블

A2A 통합 엔티티 관계도

MaintQ의 발주 초안 하나가 은행의 이체 요청으로 이어지고, 그 사이를 추적 ID가 있습니다



테이블 24개 중 A2A가 직접 건드리는 건 5개뿐입니다

나머지는 각 회사 내부에 그대로 남습니다 — 경계를 넘는 데이터를 최소로 유지했습니다

전체 지표 요약

프로젝트	지표	결과	목표	비고
MaintQ	환각률 (미지 코드 추측)	0.0%	0%	달성
MaintQ	근거 인용률	92.6%	100%	안정실패 0 — 검색 미호출 개선 중
MaintQ	안전 경고 정확도	81.5%	—	안정실패 1칸
MaintQ	부품 품번 특정률	40.0%	—	제조사 데이터 미공개 — 자동화 불가
MaintQ	평가 실행 시간	61s	—	132s에서 54% 단축 (지표는 분리 측정)
InsuQ	Hit@5 (실손, HyDE 적용)	0.9167	0.85	기준선 0.8333 대비 개선
InsuQ	Hit@5 (화재재물 기준선)	0.6346	—	HyDE 적용 시 0.7907 (시간 초과로 보류)
InsuQ	거부 정확도 / 오탐율	100% / 0%	≥90%	초과 달성
InsuQ	과잉 거부율 (화재 트랙)	0.0577	—	0.1154에서 개선 (토큰 예산 상향)
InsuQ	응답 p95	10.4s	≤30s	모델 교체 전 44.7s
InsuQ	채점기 ↔ 사람 일치율	0.9167	≥0.80	일치율 검증 후에만 지표 인정
FinAllQ	자동 테스트 / 회귀	1,500+ / 0	—	셔플 + 100회 반복으로 우연성 제거
FinAllQ	목록 조회 응답 시간	-68%	—	N+1 제거
FinAllQ	갱신 유실 / 부정 승인	0건 / 0건	0	비교 후 교체 + 직무 분리

측정 규율 - 개선과 운을 구분하는 법

개선인가, 운이 좋았던 것인가

같은 질문에 답이 실행마다 갈립니다. 그래서 한 번 재서 좋아진 것은 개선으로 인정하지 않습니다

같은 문항을 3회 반복한다

안정 실패

X

X

X

3회 모두 실패
1·2·3회 전부 X

흔들림

X

0

X

실행마다 갈림
결과가 매번 다름

안정 통과

0

0

0

3회 모두 통과
1·2·3회 전부 0

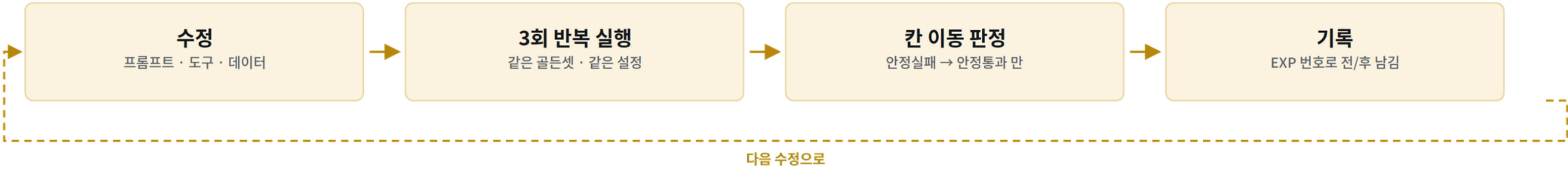
개선으로 인정하는 조건

안정 실패 → 안정 통과

이 칸으로 넘어간 것만 개선입니다

흔들림 → 통과 는 인정하지 않습니다

원래 가끔 되던 것이 이번에 됐을 뿐일 수 있습니다.
반대로 통과 → 흔들림도 회귀로 세지 않습니다.



원인을 제거하지 못하면, 최소한 속지는 않게 만듭니다

도구 루프의 비결정성은 외부 원인이라 제거하지 못했습니다. 대신 측정 규칙을 바꿔 개선과 운을 구분했습니다

멀티홉(assess-loan → verify-collateral-insurance)

① MaintQ → FinAllQ 요청

```
json

POST /a2a/skills/assess-loan
Authorization: Bearer <FINALLQ_SERVICE_TOKEN>
X-A2A-Partner-Id: maintq-agent

{
  "requester": {
    "finallq_company_id": "FQ-1043",
    "building_id": "BLD-A"
  },
  "request_chain_id": "chain-8f2a4c19",
  "loan_amount": 5000000000,
  "purpose": "노후 설비 교체 - 인버터 라인 2호기",
  "collateral_building_id": "BLD-A"
}
```

② FinAllQ → InsuQ 2차 홉

```
json

POST /a2a/skills/verify-collateral-insurance
Authorization: Bearer <INSUQ_SERVICE_TOKEN>
X-A2A-Partner-Id: finallq-agent

{
  "requester": {
    "finallq_company_id": "FQ-1043",
    "building_id": "BLD-A"
  },
  "request_chain_id": "chain-8f2a4c19",
  "building_id": "BLD-A",
  "required_coverage": 5000000000
}
```

③ InsuQ 회신

```
json

200 OK

{
  "status": "completed",
  "policy_valid": true,
  "coverage_amount": 3000000000,
  "insured_value": 5000000000,
  "sufficient": false,
  "evidence": ["재산종합보험 보통약관 제12조, p.31"]
}
```

④ FinAllQ → MaintQ 최종 회신

```
200 OK

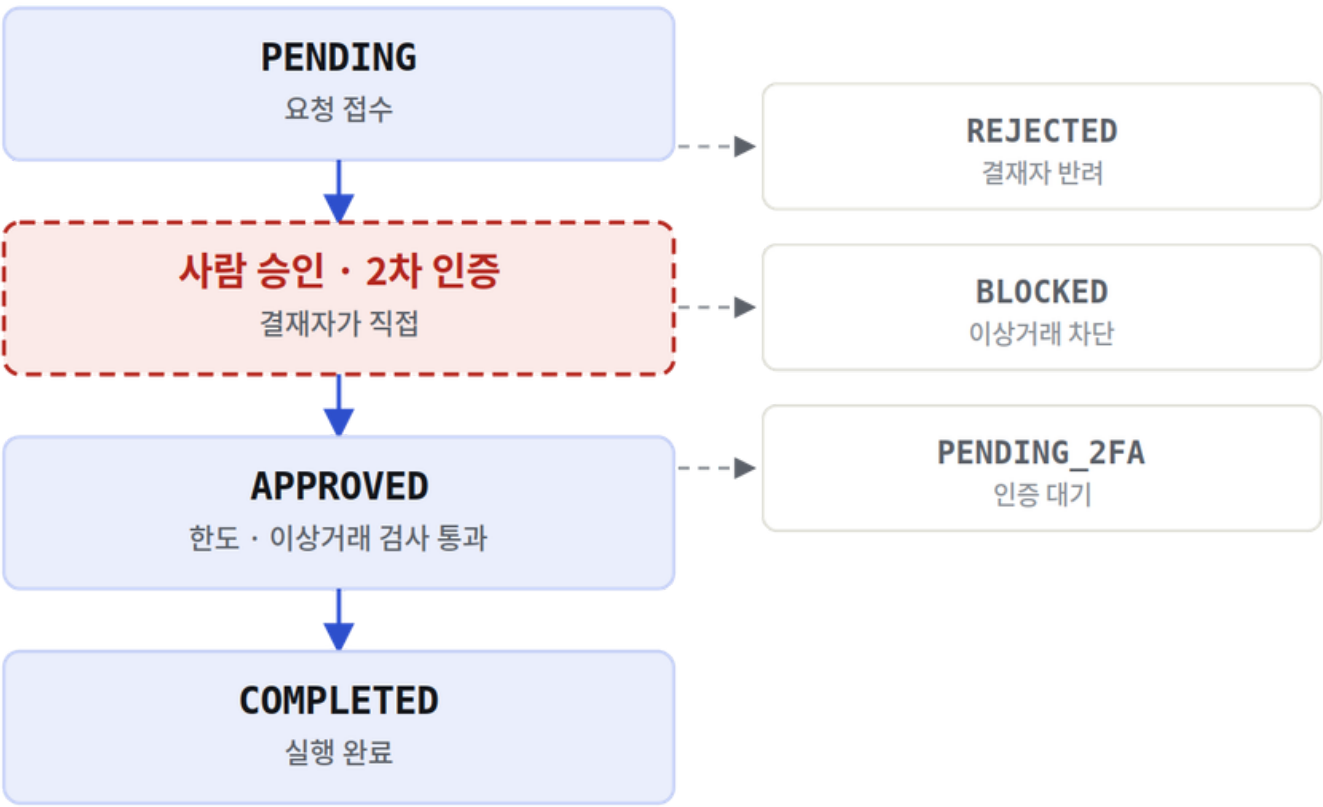
{
  "status": "completed",
  "decision": "conditional",
  "condition_note": "보험 3억 → 5억 증액 필요",
  "collateral_check": {
    "coverage_amount": 3000000000,
    "sufficient": false
  },
  "market_context": null
}
```

상태 전이 : 자동 거절은 있고 자동 승인은 없다

돈이 움직이는 두 도메인의 상태 전이

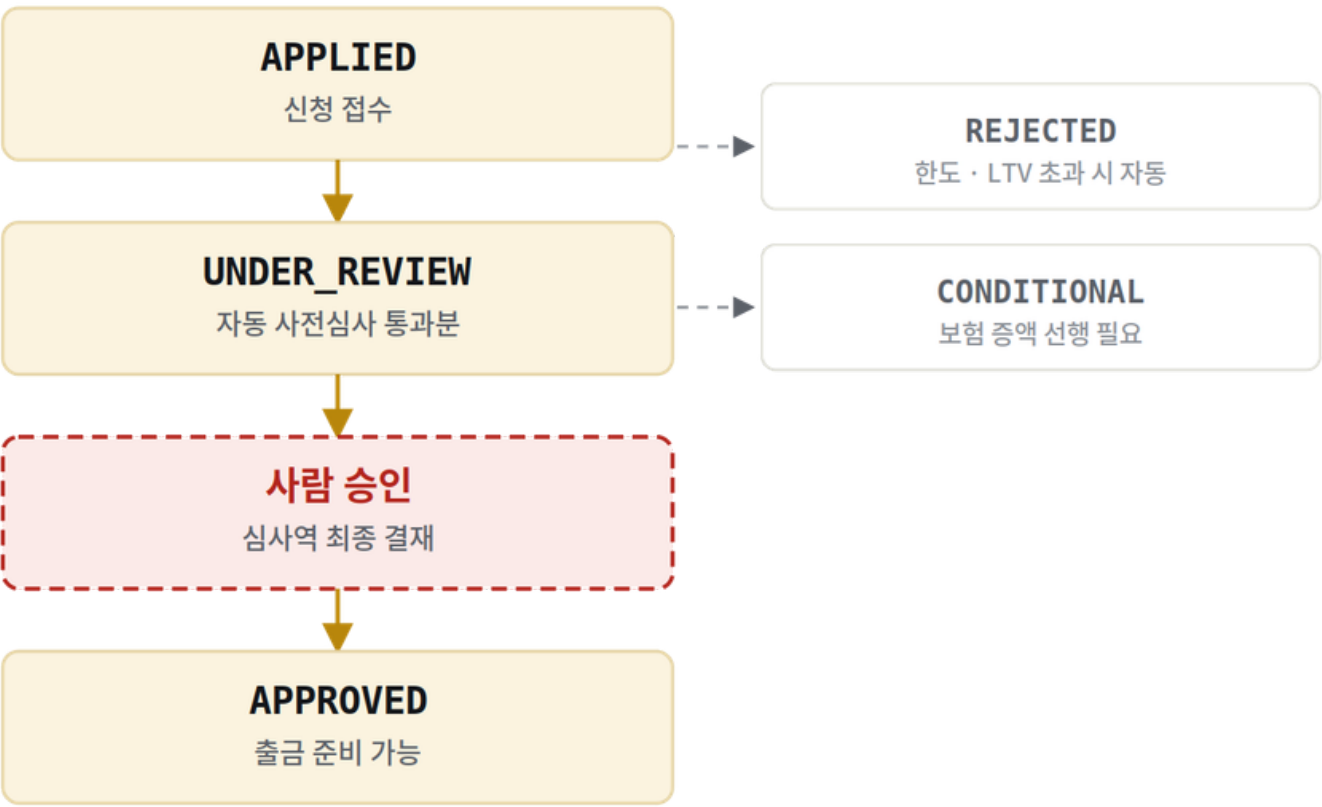
어느 쪽도 시작 상태에서 완료 상태로 바로 갈 수 없습니다. 사람 승인을 거치는 간선만 존재합니다

이체 (Transfer)



PENDING → COMPLETED 직행 간선 없음

여신 (Loan)



자동 거절은 있으나 자동 승인 간선은 없음

자동 거절은 있고, 자동 승인은 없습니다

거절은 되돌릴 수 있고 승인은 되돌리기 어렵습니다 — 비대칭이 설계 의도입니다

시연 영상

