인사이트

테스트가 전부 초록인데 왜 사용자가 먼저 발견했을까

방동걸 (주)웨이스 대표이사 2026년 8월 23일

전에 저희는 이 블로그에 합의는 검증이 아니다라고 썼습니다. 같은 모델을 쓰는 에이전트 스무 개는 같은 오류에 만장일치로 동의하므로, 검증은 언제나 시스템 밖의 증거가 한다는 이야기였습니다. 웨이스가 VEXPLOR를 개발하며 그 원칙을 직접 다시 확인하게 된 일을 이번 글에 적습니다.

원칙은 그 글에서 끝났는데, 정작 어려운 것은 그다음이었습니다. **시스템 밖의 증거를 실제로 어떻게 만드느냐**입니다. 저희는 검사를 짰다고 생각했는데 그 검사가 대상에 닿지 않는 일을 최근 한 주 동안 아홉 번 겪었습니다. 하루에 세 번 나온 날도 있었습니다.

공통점이 하나 있었습니다. **전부 초록이었습니다.**

이 글은 그 아홉 건을 네 가지로 갈라 정리한 것입니다. 원칙이 아니라 구성에 대한 이야기입니다.

01

초록불은 "대상이 옳다"를 뜻하지 않습니다

검사가 초록이라는 것은 정확히 한 가지만 말합니다. **검사가 실행됐고, 검사가 본 것이 기대와 같았다.** 여기에는 두 개의 가정이 숨어 있습니다. 검사가 실제로 실행됐다는 것, 그리고 검사가 본 것이 우리가 판정하려던 그 대상이라는 것입니다.

저희가 겪은 아홉 건은 전부 이 두 가정 중 하나가 조용히 깨진 경우였습니다. 조용히 깨졌다는 표현이 중요합니다. 가정이 깨지면 검사는 빨간불이 되는 게 아니라 **초록불인 채로 무의미해집니다.**

유형	깨진 가정	겉으로 보이는 것
대상에 안 닿음	검사가 본 것이 다름	초록
전제를 자기가 만듦	검사 환경이 오염	초록
대리 지표를 봄	본 것이 결과가 아님	초록
대상에게 물어봄	관측이 대상을 변형	초록



네 가지 모두 겉으로는 초록입니다. 무엇이 깨졌는지가 다를 뿐입니다.

02

첫째, 검사가 대상에 아예 닿지 않았습니다

가장 흔했고 가장 오래 숨었습니다.

한 배포 도구에서 매니페스트를 읽는 함수가 훅 목록을 배열로 반환하는데, 그 값을 받는 판정 함수는 매니페스트 객체를 기대하고 있었습니다. 배열을 넘기면 판정은 언제나 "권한 없음"을 반환합니다. 그런데 그 경로는 실패가 아니라 **아무 경고도 표시하지 않는 정상 상태**로 취급되고 있었습니다. 어느 PC에서도 그 경고는 영영 뜨지 않았을 텐데, 테스트는 통과했습니다. 검사가 "경고가 안 뜬다"를 확인하고 있었기 때문입니다.

또 하나는 성격이 조금 다릅니다. 원본 데이터에서 필요한 것을 뽑아 내는 변환 단계가 객체의 한 필드만 가져가고 **나머지 개념 여덟 종을 통째로 버리고 있었습니다.** 화면 정의도, 프로세스도, 관계도 전부 빠졌는데 파이프라인은 끝까지 정상으로 돌았습니다. 버린 것은 예외를 던지지 않기 때문입니다. **누락은 오류로 보이지 않고 그냥 "적은 결과"로 보입니다.** 그래서 이 유형은 "결과가 나왔는가"가 아니라 "들어간 것 대비 나온 것이 몇인가"를 세야 잡힙니다.

같은 종류로 세 건이 더 있었습니다. 타입 검사 설정에서 애플리케이션 디렉터리가 대상 목록 밖에 있어 실제 오류 두 건이 초록불 아래 있었던 일, 품질 게이트가 존재하지 않는 스크립트 이름을 부르고 있었던 일, 데이터 대조 검사가 권한 제한이 걸린 계정으로 대조군을 세어 양쪽 다 0이 나온 일입니다. **0과 0을 비교하면 언제나 일치합니다.**

03

둘째, 검사가 자기 전제를 스스로 만들었습니다

빌드 산출물을 읽는 테스트가 있었습니다. 그런데 실행 순서상 테스트가 빌드보다 먼저 돌았습니다. 그런데도 로컬에서는 늘 초록이었습니다. **개발자 PC에 지난번 빌드 결과가 남아 있었기 때문입니다.**

이 결함은 깨끗한 환경에서 처음 돌렸을 때 바로 드러났습니다. 문제는 그 깨끗한 환경이 오래 없었다는 것입니다.

비슷한 사례가 하나 더 있습니다. 지속적 통합이 열한 번 연속 빨간불이었는데 아무도 몰랐습니다. 원인은 실행 컨테이너에 런타임이 없어 첫 단계부터 죽은 것이었고, 그래서 **품질 게이트 단계는 단 한 번도 실행된 적이 없었습니다.** 안 도는 것과 돌면서 실패하는 것은 화면에서 다르게 보이지만, 아무도 그 화면을 안 보면 결과가 같습니다.

여기서 한 걸음 더 들어간 사례가 있습니다. 어떤 가드를 사흘 동안 네 번 고쳤는데, **그 가드가 맞는지를 한 번도 재지 않았습니다.** 테스트 스위트 열아홉 개가 있었지만 내용을 열어 보니 전부 이미 관측된 오탐과 이미 잡히던 경로뿐이었고, **가드가 실제로 막아야 할 적대적 경로는 0건**이었습니다. 그래서 오탐을 없애려던 수정이 새 오탐을 만들었는데도 열아홉 개가 전부 녹색으로 나왔고, 그 녹색이 수정을 통과시켰습니다.

테스트 개수는 안전의 지표가 아닙니다. **스위트가 무엇을 담고 있지 않은지**가 지표입니다.

04

셋째, 검사가 결과 대신 대리 지표를 봤습니다

이건 가장 뼈아팠습니다.

문서 발행 파이프라인에서 마크다운을 편집기 문서 모델로 변환할 때, 표의 첫 헤더 셀이 비어 있으면 빈 문자열 텍스트 노드가 생깁니다. 그 문서 모델의 스키마에서 빈 문자열 텍스트 노드는 유효하지 않습니다. 결과는 **본문 전체가 렌더되지 않는 것**입니다.

그런데 API는 201을 반환했습니다. 제목도 정상, 소셜 미리보기 이미지도 200. 저희 검증은 상태 코드와 이미지 응답을 확인하고 있었으므로 전부 통과했습니다. 본문이 통째로 사라진 글이 라이브에 올라갔고, **그걸 발견한 사람은 검사가 아니라 그 글을 읽으러 들어간 사람이었습니다.**

같은 날 또 하나 있었습니다. 배포 직후 페이지를 확인했더니 구버전이 보였습니다. 캐시였습니다. 캐시를 우회해 다시 부르니 정상이었습니다. 처음 응답만 믿었으면 배포 실패로 오판하고 되돌렸을 것입니다.

상태 코드는 "요청이 처리됐다"이지 "결과가 옳다"가 아닙니다. 200은 서버가 무언가를 돌려줬다는 뜻이고, 그 무언가가 빈 페이지여도 200입니다.

05

넷째, 검사자가 검사 대상에게 물어봤습니다

앞의 셋은 코드 문제였지만 이건 설계 문제입니다.

같은 주에 일본어 페이지를 배포하고 화면을 봤는데 한국어로 나왔습니다. 번역 계층 결함을 의심했습니다. 그런데 서버 응답을 직접 받아 보니 일본어 원문 그대로였고 한국어는 한 글자도 없었습니다. **브라우저의 자동 번역이 화면을 바꾼 것이었습니다.**

단서는 화면에 뜬 단어가 우리 한국어 원고에 없는 말이라는 점이었습니다. 우리는 "가격"이라고 쓰는데 화면은 "수수료"였습니다. 관측 도구가 대상을 변형해 놓고 그 변형된 것을 보여 주고 있었던 것입니다.

이것은 AI에게 답을 되물어 확인하는 것과 같은 구조의 오류입니다.

여기서 최근 연구 하나가 정확히 겹칩니다. 하버드 경영대학원 연구진이 컨설턴트 70여 명이 AI와 나눈 대화 기록을 분석했더니, **사람이 사실을 확인하고 반박할수록 모델이 물러나는 대신 설득의 강도를 높였습니다.** 연구진은 이 현상에 이름을 붙이고, 사람이 하던 방식의 반대신문이 이 상대에게는 맞지 않는 도구라고 결론지었습니다. 그리고 필요한 것은 사람이 더 잘 보는 것이 아니라 **별도의 감독 장치를 설계하는 것**이라고 적었습니다.

같은 연구팀의 앞선 실험에서는 컨설턴트 758명에게 AI가 잘 풀지 못하는 문제를 주자, AI를 쓴 쪽의 정답률이 AI를 안 쓴 쪽의 84.5%에서 60~70%로 떨어졌습니다. 참가자는 전원 그 분야 전문가였고 문제도 그들의 전문 영역 안이었습니다. 여기서 흔한 오해를 하나 짚어 두겠습니다. **이 실험의 경계는 AI의 능력 경계이지 사람의 전문성 경계가 아닙니다.** 전문가라서 걸러낸 것이 아니라, 전문가인데도 못 걸렀습니다.

06

그래서 검사를 어떻게 다시 짰나

네 가지를 바꿨습니다. 대단한 것은 없고 전부 "검사가 자기 자신을 검사하게" 만드는 쪽입니다.

- **검사가 대상에 닿았는지를 검사한다.** 대조군이 0이면 통과가 아니라 실패로 봅니다. 비교 대상의 개수를 세어 0이면 그 자리에서 멈추게 했습니다. 0과 0의 일치는 정보가 아닙니다.
- **음성 단정을 함께 쓴다.** "있어야 할 것이 있다"만 검사하면 절반입니다. **"없어야 할 것이 없다"**를 같이 넣습니다. 특정 경고가 이 환경에서 뜨면 그것이 결함이라는 단정을, 뜨지 않는 것이 정상인 환경에 심어 둡니다.
- **대리 지표 대신 결과를 센다.** 발행 후 상태 코드가 아니라 실제 문서에서 문단·소제목·표의 개수를 세고, 본문 키워드 서너 개를 직접 찾습니다. 0이면 렌더 실패입니다.
- **관측 도구를 대상에서 떼어 낸다.** 화면으로 배포물을 판정하지 않고 서버 응답을 직접 받습니다. AI 출력은 같은 대화 안에서 확인하지 않고 독립 수단으로 대조합니다.

마지막 항목이 가장 중요합니다. **검증하는 계층과 검증받는 계층을 같은 자리에 두지 않는 것**이 네 유형 전부의 공통 해법입니다.

대응	막는 유형	실제 검사 문장
대상 도달 단정	안 닿음	대조군 0이면 실패
음성 단정	조용한 통과	뜨면 결함인 것 명시
결과 계수	대리 지표	문단·표 개수를 센다
관측 분리	관측이 변형	서버 응답을 직접



막는 쪽에 답을 내는 쪽으로 이어지는 연결이 없습니다. 스스로 완화할 수 없어야 검사가 헐거워지지 않습니다.

07

공장에서는 이 문제가 더 무겁습니다

소프트웨어에서는 빈 페이지가 올라가고 사람이 발견합니다. 설비에서는 판단이 그대로 동작으로 나갑니다.

현장 시스템의 검사도 똑같이 이 네 유형에 걸립니다. 태그를 못 읽고 있는데 값이 0으로 들어와 정상 범위로 판정되는 경우가 첫째 유형이고, 설비가 멈춘 구간을 빼지 않고 가동률을 계산해 늘 초록인 것이 셋째

유형입니다. 특히 **통신이 끊겼는데 마지막 값이 유지되는 상태**는 어느 대시보드에서도 빨간불이 아닙니다. 값이 있기 때문입니다.

그래서 저희는 판단이 설비로 나가는 자리에 확정적인 계층을 따로 둡니다. 확률적으로 답을 내는 계층과, 규칙으로 막는 계층을 갈라 놓고, **막는 쪽은 답을 내는 쪽이 스스로 고칠 수 없게** 합니다. 자기가 만든 검사를 자기가 완화할 수 있으면 그 검사는 시간이 지나면서 반드시 헐거워집니다.

08

한 줄로 줄이면 이렇습니다

초록불은 "검사가 통과했다"이지 "대상이 옳다"가 아닙니다.

검사를 새로 짤 때 저희가 지금 묻는 질문은 하나로 줄었습니다. 이 검사가 틀린 것을 정말로 볼 수 있는 자리에 있는가. 답이 애매하면, 그 검사는 초록이어도 아직 아무것도 보증하지 않습니다.

WACE(웨이스)는 공장이 스스로 판단하고 멈추지 않게 하는 제조 AI 운영체제 VEXPLOR를 만드는 회사다.

원문 blog.vexplor.com/post/green-light-does-not-mean-correct
발행 2026년 8월 23일 · 방동걸 (주)웨이스 대표이사

회사 주식회사 웨이스 (WACE Inc.) · 대표이사 방동걸 · 사업자등록번호 439-87-02852
주소 (18462) 경기 화성시 동탄구 동탄영천로 150, 에이동(A동) 1702-1708호 (영천동, 실리콘앨리동탄)
연락처 전화 070-4730-0629 · contact@wace.me
홈페이지 wace.me · linkedin.com/company/wace-inc

원문 홈페이지 링크드인