MUMULAB

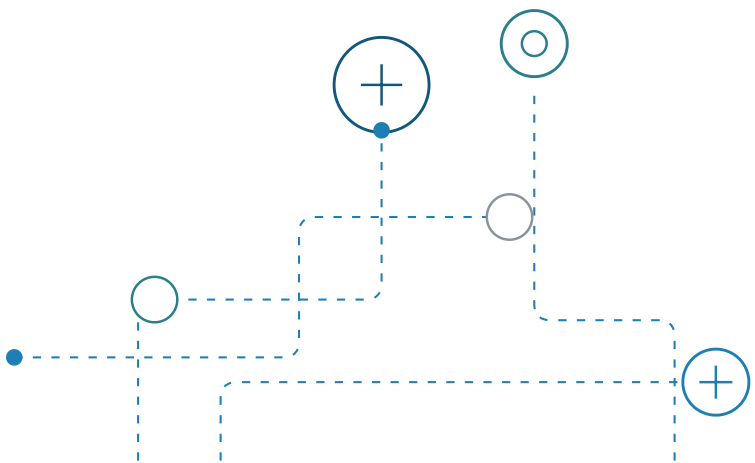
INSIGHT SERIES

2026-08-15

AI의 미래는 공장에 있었다

Lean Manufacturing으로 이해하는 Agentic AI와 지식노동의 새로운 생산시스템

MUMULAB



Contents

01

AI의 최신 용어가 공장에서 익숙한 이유

Agentic AI를 기술 유행이 아니라 인간과 기계의 협업 시스템으로 다시 본다.

AI를 공부하기 시작하면 낯선 단어가 줄을 선다. Agentic AI, Human-in-the-loop, Orchestration, Guardrail, Eval, Observability, Harness Engineering. 어제까지 없던 미래의 언어처럼 들린다. 그런데 제조업의 생산시스템을 오래 들여다본 사람에게는 이상하리만큼 익숙하다. 단어는 새롭지만 질문의 모양이 오래되었기 때문이다.

공장은 이미 수십 년 동안 같은 질문을 풀어왔다. 무엇을 사람이 해야 하는가. 무엇을 기계가 해야 하는가. 언제 자동으로 다음 단계로 넘어가야 하는가. 이상이 생기면 어디에서 멈춰야 하는가. 멈춘 뒤에는 누구에게 알려야 하는가. 한 번 해결한 문제가 다시 생기지 않도록 작업표준을 어떻게 바꿀 것인가.

제조업이 물리적 작업의 Production System을 만들었다면, Agentic AI는 지식노동의 Production System을 만들고 있다.

MUMULAB

이 책의 주장은 "AI와 공장은 같다"가 아니다. 둘이 해결하려는 **운영 문제의 원형이 닮았다**는 것이다. 제조업이 물리적 작업의 Production System을 만들었다면, Agentic AI는 지식노동의 Production System을 만들고 있다. 제조업은 소재와 설비와 작업자의 흐름을 설계했다. 이제 기업은 문서와 데이터와 판단과 AI 에이전트의 흐름을 설계해야 한다. 대상은 달라도, 좋은 흐름·품질·예외관리·지속개선이라는 질문은 이어진다.

1.1 가장 먼저 바꿔야 할 질문

AI 도입 회의에서는 보통 이렇게 묻는다.

> "어떤 일을 AI로 대체할 수 있을까?"

이 질문은 도구에서 시작한다. 그래서 답도 "메일을 써준다", "보고서를 요약한다", "PPT를 만든다" 같은 기능 목록으로 끝나기 쉽다. Lean의 관점에서는 출발점이 다르다.

> "고객에게 가치가 전달되는 전체 흐름을 어떻게 다시 설계할 것인가?"

이 질문을 던지면 업무가 보인다. 왜 이 일을 하는지, 어디에서 기다리는지, 어떤 정보를 다시 입력하는지, 누가 같은 내용을 세 번 검토하는지, 무엇이 틀리면 큰 사고가 되는지 드러난다. 그다음에야 사람, AI, 규칙, 시스템의 역할을 나눌 수 있다.
Toyota는 TPS를 설명하면서 Jidoka와 Just-in-Time을 두 축으로 두고, 이상이 생기면 즉시 멈춰 불량이 다음 공정으로 흐르지 않게 하는 구조를 강조한다. 더 흥미로운 대목은 자동화의 순서다. 먼저 사람이 일을 정확히 해내고, 낭비와 불균형을 줄여 반복 가능한 공정을 만든 뒤 기계로 옮긴다. 엉킨 일을 그대로 자동화하면 엉킴만 빨라진다는 뜻이다.

1.2 이 책에서 쓰는 네 단어

> **처음 읽는 독자를 위한 번역**
>
> **AI 에이전트**는 목표를 받아 여러 단계를 계획하고 도구를 사용해 결과까지 수행하는 AI 프로그램이다. **워크플로**는 미리 정한 순서대로 움직이는 업무 흐름이다.
> **가드레일**은 금지 규칙, 권한, 형식 검사처럼 위험한 행동을 막는 안전장치다. **평가(Eval)**는 AI가 실제로 일을 잘했는지 반복해서 시험하는 테스트다.

Anthropic은 워크플로와 에이전트를 구분한다. 워크플로는 코드로 정해 둔 길을 따른다. 에이전트는 상황에 따라 스스로 경로와 도구 사용을 결정한다. 자율성이 커질수록 유연해지지만 비용과 오류의 연쇄 가능성도 커진다. 따라서 모든 업무를 자율 에이전트로 바꾸는 것이 성숙의 증거는 아니다. 필요한 만큼만 복잡하게 만드는 것이 오히려 좋은 설계다.

이제 공장의 언어를 빌려 세 가지를 살펴보자. 첫째, 사람과 AI의 작업조합. 둘째, 이상을 발견하고 멈추는 Digital Jidoka. 셋째, 좋은 결과가 반복되도록 환경을 설계하는 Harness다. 마지막에는 이 비유가 어디에서 깨지는지도 분명히 짚는다.

1.3 핵심 요약

- AI Transformation의 단위는 개별 도구가 아니라 고객가치를 만드는 전체 업무 흐름이다.
- 자동화 전에 현재 업무를 관찰하고 반복 가능한 표준을 만들어야 한다.
- 에이전트의 자율성보다 역할, 품질기준, 정지조건을 먼저 설계해야 한다.
- Lean은 답안지가 아니라, AI 시스템을 설계할 때 쓸 수 있는 강력한 질문의 체계다.

02

AI 에이전트는 혼자 일하는 로봇이 아니다

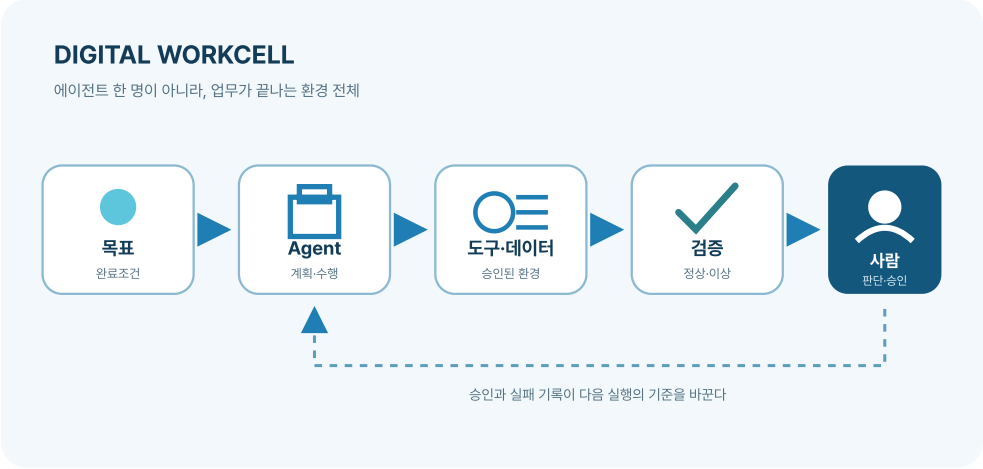
에이전트, 워크플로, 도구, 검증, 인간 승인을 한 묶음으로 이해하는 가장 쉬운 방법.

AI 에이전트를 설명하는 그림에는 종종 똑똑한 로봇 하나가 등장한다. 사람은 목표만 말하고 로봇이 검색하고 판단하고 실행한다. 이해하기 쉬운 그림이지만 실제 기업 업무를 설명하기에는 부족하다. 로봇 한 명이 아니라 **작업환경 전체**가 성과를 만들기 때문이다.

2.1 에이전트와 디지털 워크셀

제조현장의 Workcell은 기계 한 대를 뜻하지 않는다. 작업자, 설비, 치공구, 자재, 작업표준, 품질기준, 이상 대응이 하나의 셀을 이룬다. 같은 관점으로 보면 AI 에이전트 단독을 Digital Workcell이라고 부르기는 어렵다.

Agent + Tool + Data + Validation + Human Approval이 연결되어야 비로소 하나의 Agentic Workflow, 즉 디지털 워크셀에 가까워진다.



AI 에이전트 단독이 아니라 목표·도구·데이터·검증·인간 승인이 연결된 디지털 워크셀 · MUMULAB

표 2-1. 에이전트와 디지털 워크셀의 차이

구분	에이전트 단독	디지털 워크셀
목표	질문에 답하거나 과업 수행	고객가치와 완료조건 달성
입력	프롬프트, 파일	승인된 데이터와 업무 상태
실행	모델이 계획·도구 사용	에이전트·도구·규칙이 역할 분담
품질	답변이 그럴듯한지 확인	실제 결과와 시스템 상태를 검증
예외	오류 메시지	중단·알림·사람 승인·복구 절차
개선	프롬프트 수정	실패 원인을 표준과 평가에 반영

자료: MUMULAB 정리

2.2 월요일 아침 경쟁사 뉴스 보고서

직장인의 익숙한 업무를 예로 들어보자. 매주 월요일 아침, 담당자는 경쟁사와 산업 뉴스를 찾아 팀장에게 1페이지 보고서를 보낸다.

처음에는 AI에게 "지난주 주요 뉴스를 요약해줘"라고 시킨다. 빠르지만 출처가 빠지거나 오래된 기사와 최신 기사가 섞일 수 있다. 숫자는 그럴듯한데 원문에 없을 수도 있다. 이것은 좋은 모델을 쓴 **단발성 AI 사용**이다.

디지털 워크셀은 다르게 설계한다.

1. 승인된 언론사와 공시에서 지난 7일의 자료만 가져온다.

2. 동일 사건을 묶고 원문 날짜와 출처를 보존한다.

3. AI는 사실, 해석, 추정을 서로 다른 필드에 작성한다.

4. 수치와 인용문은 원문 대조 검사를 통과해야 한다.

5. 회사에 미치는 영향과 투자 관점은 근거와 함께 초안을 만든다.

6. 외부 공유 전에는 담당자가 최종 승인한다.

7. 틀린 요약은 실패 사례로 저장해 다음 평가 항목에 추가한다.

여기서 AI는 중요한 부품이지만 전부가 아니다. 어떤 출처를 믿을지, 완료를 무엇으로 볼지, 무엇이 틀리면 멈출지, 누가 승인할지가 함께 설계되어야 한다.

> **쉬운 판별법**
>
> AI가 "완료했습니다"라고 말한 것과 업무가 실제로 완료된 것은 다르다. 메일 초안을 썼다는 문장이 아니라 발신함에 승인된 메일이 존재하는지, 표에 숫자를 적었다는 답변이 아니라 원문과 숫자가 일치하는지를 확인해야 한다.

Anthropic의 평가 가이드는 이를 **결과(outcome)**와 **경로(trace)**로 나누어 본다. 결과는 실제 환경에서 무엇이 바뀌었는가다. 경로는 에이전트가 어떤 도구를 어떤 순서로 사용했는가다. 좋은 운영은 둘 다 본다. 결과만 보면 우연히 맞은 위험한 경로를 놓치고, 경로만 보면 실제 업무가 끝나지 않았는데도 성공으로 오판할 수 있다.

2.3 자율성은 목표가 아니라 설정값이다

업무마다 필요한 인간 개입의 위치는 다르다.

- **Human-out-of-the-loop**: 되돌리기 쉽고 위험이 낮은 분류, 파일명 정리, 초안 생성.
- **Human-on-the-loop**: 대시보드로 감시할 수 있고 문제가 생기면 빠르게 중단 가능한 반복 업무.
- **Human-in-the-loop**: 돈, 사람, 규정, 대외발신처럼 결과가 크거나 되돌리기 어려운 결정.

성숙한 조직은 모든 업무를 한 방향으로 밀지 않는다. 위험과 가역성에 따라 자율성 수준을 다르게 둔다. Microsoft의 Responsible AI 가이드도 사람·금전·규정에 영향을 주거나 되돌리기 어려운 행동에는 인간 승인을 두고, 출시 뒤에는 성능 저하와 드리프트를 계속 관찰하라고 권한다.

따라서 "사람이 루프에 있으면 덜 발전한 시스템"이라는 생각은 위험하다. 브레이크가 달린 자동차가 덜 발전한 자동차가 아닌 것과 같다. 중요한 것은 사람을 몇 번 클릭하게 하는가가 아니라, **사람의 판단이 가장 가치 있는 순간에 배치되어 있는가**다.

03
사람과 AI의 작업조합표

무엇을 AI에게 맡길지가 아니라 언제 누구에게 일을 넘길지를 설계한다.

Lean의 Standardized Work Combination Table은 사람의 수작업 시간, 이동 시간, 설비의 자동가공 시간을 하나의 시간축에 놓는다. 목적은 사람을 쉬지 못하게 만드는 것이 아니다. 작업자와 기계가 언제 기다리고, 언제 겹치며, 어디에서 병목이 생기는지 보이게 하는 것이다.

지식노동에도 같은 시선이 필요하다. 현재 대부분의 AI 사용은 "사람이 질문하고 AI가 답한다"는 대화 단위로 측정된다. 하지만 실제 업무는 검색, 판단, 입력, 승인, 발신, 기록까지 이어진다. 챗봇의 답변 속도가 빨라도 사람이 복사하고 다시 확인하고 다른 시스템에 입력하느라 20분을 쓴다면 전체 흐름은 좋아지지 않았다.

3.1 작업을 세 종류로 나눈다

Human-Agent Work Design의 첫 단계는 사람과 AI의 강점을 추상적으로 논하는 것이 아니다. 실제 업무를 작은 행동으로 쪼개고 세 종류로 배치하는 것이다.

표 3-1. Human-Agent Work Design의 기본 역할

역할	잘 맞는 일	예시
사람	목적, 맥락, 가치판단, 책임	무엇이 중요한 뉴스인지 결정, 대외발신 승인
AI	대량 탐색, 변환, 비교, 초안	기사 수집, 중복 제거, 요약, 형식 변환
규칙·시스템	확정적 검사와 기록	날짜 범위, 필수필드, 금액 한도, 로그 저장

자료: MUMULAB 정리

여기서 자주 생기는 실수는 모든 판단을 AI에게 넣는 것이다. 날짜 형식, 필수 항목, 금액 한도처럼 명확한 규칙은 일반 코드가 더 빠르고 안정적이다. AI는 모호한 언어와

비정형 자료를 다루는 데 강하지만, 같은 입력에도 다른 출력을 낼 수 있다. 확정적 규칙과 확률적 판단을 분리해야 한다.

3.2 인계점이 성능을 결정한다

좋은 조합은 누가 더 똑똑한지보다 **어디에서 일을 넘기는지**로 결정된다. 월요일 뉴스 보고서라면 다음과 같이 설계할 수 있다.

사람: 관찰할 주제와 독자를 정의
시스템: 기간 · 출처 조건으로 자료 수집
AI: 사건을 묶고 초안 작성
시스템: 링크 · 날짜 · 숫자 형식 검사
사람: 영향도와 표현을 판단하고 승인
시스템: 발행 · 보관 · 성과 기록

사람이 처음부터 모든 기사를 읽는 낭비는 줄지만, 의미를 판단하는 책임은 남는다. AI는 대량의 자료를 좁히고 초안을 만드는 데 집중한다. 시스템은 AI에게 맡길 필요가 없는 명확한 검사를 담당한다.

> **자동화의 숨은 재공품**
>
> 공장의 WIP가 공정 사이에 쌓인 재공품이라면, 사무실의 WIP는 읽지 않은 메일, 검토 대기 문서, 미승인 초안, 너무 긴 대화 맥락이다. 에이전트를 여러 개 붙이면 처리량보다 '검토 대기'가 먼저 늘 수 있다.

에이전트 수를 늘리는 것은 라인에 설비를 추가하는 것과 비슷하다. 앞뒤 공정의 속도가 맞지 않으면 병목만 이동한다. 에이전트 A가 100개의 초안을 만들고 사람이

하루에 10개만 검토할 수 있다면 90개는 가치가 아니라 재공품이다. 동시 작업 수, 승인 큐, 컨텍스트 크기를 관리해야 하는 이유다.

3.3 Standard Work는 AI를 가두는 문서가 아니다

Lean Enterprise Institute는 표준작업을 개선의 기준선으로 설명한다. 표준이 없으면 어제와 오늘의 차이를 알 수 없고, 좋아졌는지도 판단할 수 없다. AI 지침과 SOP도 같다.

좋은 표준은 "항상 똑같이 써라"가 아니다. 목적, 입력, 허용 도구, 완료조건, 금지행동, 예외처리를 명확히 한다. 그리고 실패가 발견되면 바뀐다. 표준은 창의성을 없애는 울타리가 아니라, 반복되는 실수를 제거해 창의적 판단에 시간을 돌려주는 바닥이다.

AI의 생산성은 답변 한 번의 속도가 아니라, 고객가치가 끝까지 흐르는 시간으로 측정해야 한다.

AI의 생산성은 답변 한 번의 속도가 아니라, 고객가치가 끝까지 흐르는 시간으로 측정해야 한다. 따라서 성과지표도 바뀌어야 한다. 생성된 문서 수보다 완료까지 걸린 시간, 재작업률, 사람의 검토시간, 오류의 유출률, 예외 해결시간을 본다. 이것이 챗봇 사용량을 넘어 운영 시스템을 관리하는 방법이다.

04
잘 달리는 것보다 잘 멈추는 시스템

Jidoka의 관점에서 검증, 중단, 알림, 에스컬레이션을 하나의 품질 시스템으로 묶는다.

Toyota Production System의 Jidoka는 흔히 '자동화'로 번역되지만 핵심은 단순한 무인화가 아니다. 이상이 생겼을 때 즉시 감지하고, 불량이 다음 공정으로 흐르지 않게 멈추며, 사람이 원인을 확인하고 다시 발생하지 않도록 공정을 개선하는 구조다.
AI 시스템도 마찬가지다. 자율성이 커질수록 "얼마나 오래 혼자 일하는가"보다 "어떤 상황에서 스스로 멈추는가"가 중요해진다.

4.1 Guardrail 하나로는 부족하다

"Guardrail은 Digital Jidoka다"라는 문장은 기억하기 쉽지만 정확히는 부족하다. 가드레일은 금지어, 권한, 형식, 금액 한도 같은 경계다. Jidoka에 가까워지려면 다섯 요소가 함께 있어야 한다.

1. **검증**: 정상과 이상을 구분한다.

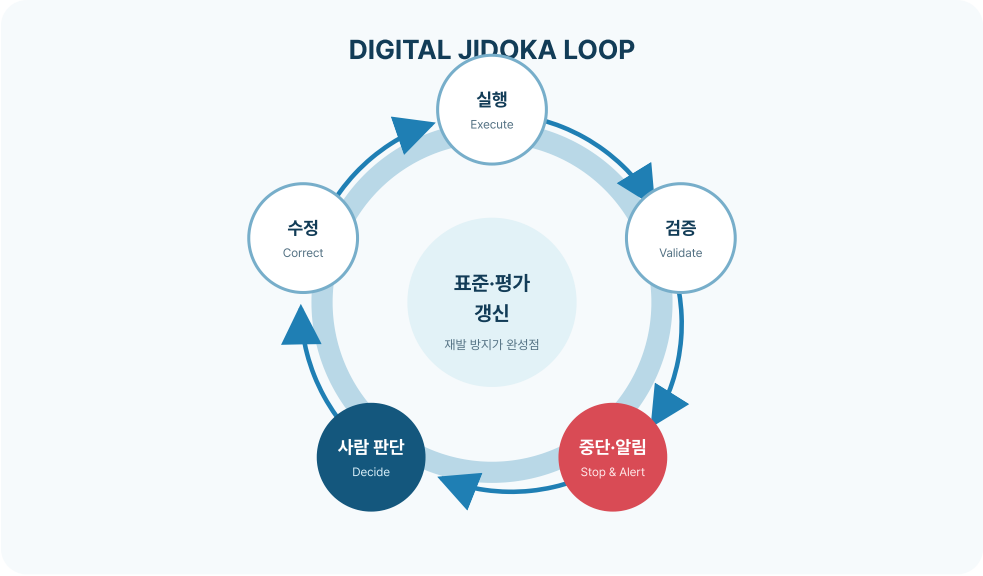
2. **중단**: 이상한 결과가 다음 단계로 넘어가지 않게 한다.

3. **알림**: 누가 봐야 하는지 즉시 전달한다.

4. **에스컬레이션**: 사람에게 필요한 정보와 선택지를 제공한다.

5. **학습**: 원인을 제거하고 표준과 평가를 갱신한다.

즉, Digital Jidoka는 `Validation → Stop → Alert → Human Decision → Correct → Update Standard`의 닫힌 고리다. 단지 위험한 단어를 막는 필터가 아니다.



검증에서 표준 갱신까지 이어지는 Digital Jidoka 루프 · MUMULAB

4.2 뉴스 보고서에서 멈춰야 할 순간

AI가 경쟁사 뉴스를 정리하는 상황을 다시 보자. 다음 조건은 자동 중단 지점이 될 수 있다.

- 핵심 수치의 원문 링크가 없다.
- 두 출처가 같은 사건의 수치를 다르게 보도한다.
- 기사의 발행일과 사건 발생일이 크게 다르다.
- '사실' 필드에 전망이나 추정 표현이 섞였다.

- 회사에 영향을 준다는 판단의 근거가 비어 있다.
- 외부 공개 문서인데 개인정보나 내부 정보가 포함됐다.

중요한 것은 "오류가 있으면 사람이 보세요"라고 막연히 넘기지 않는 것이다. 어떤 항목이 왜 실패했는지, 원문은 무엇인지, 사람이 선택할 수 있는 조치는 무엇인지 함께 보여줘야 한다. 그렇지 않으면 자동화가 사람에게 새로운 조사 업무를 떠넘긴다.

> **좋은 에스컬레이션의 형식**
>
> 중단 사유 + 영향받는 결과 + 확인할 원문 + 추천 선택지 + 재개 조건을 한 화면에 제공한다. 사람은 시스템의 추리를 처음부터 재현하는 대신, 중요한 판단에 집중할 수 있다.

4.3 정상 흐름과 예외 흐름을 함께 설계한다

AI 도입 문서는 대개 정상적인 데모만 그린다. 자료를 넣으면 요약이 나오고, 버튼을 누르면 메일이 발송된다. 실제 운영비용은 예외에서 생긴다. 출처가 닫혀 있을 때, 파일 형식이 다를 때, 권한이 없을 때, 두 규칙이 충돌할 때, 사람이 응답하지 않을 때 어떻게 할 것인가.

제조업의 Andon은 이상을 눈에 보이게 만든다. AI의 Observability도 같은 역할을 한다. 성공률 하나가 아니라 어떤 단계에서, 어떤 입력으로, 어떤 도구를 쓰다가, 어떤 규칙에 걸렸는지 추적해야 한다. 그래야 "모델이 가끔 이상하다"를 재현 가능한 문제로 바꿀 수 있다.

NIST는 배포된 AI 모니터링에서 기능, 운영, 인간요인, 보안, 규정준수, 광범위한 영향을 함께 보라고 제안한다. AI는 출시 후에도 입력 분포와 사용자 행동이 바뀌며

성능이 흔들릴 수 있다. 초기 테스트 통과는 출발점이지 품질보증의 끝이 아니다.

4.4 멈춤은 실패가 아니라 품질 기능이다

사람들은 자동화가 멈추면 시스템이 나쁘다고 느낀다. 그러나 기준 미달 결과를 조용히 발송하는 시스템보다, 이유를 설명하며 멈추는 시스템이 훨씬 성숙하다.

좋은 자동화는 한 번도 멈추지 않는 시스템이 아니다. 틀린 결과가 고객에게 닿기 전에, 정확한 이유로 멈추는 시스템이다. 저위험 업무에는 느슨한 경계를, 큰 금액·인사·대외발신에는 강한 경계를 둔다. 모든 곳에 같은 검토를 붙이면 흐름이 막히고, 어디에도 검토를 두지 않으면 사고가 흐른다. 위험 기반으로 정지조건을 설계하는 것이 핵심이다.

좋은 자동화는 한 번도 멈추지 않는 시스템이 아니다. 틀린 결과가 고객에게 닿기 전에, 정확한 이유로 멈추는 시스템이다.

05
‘알아서 잘해’가 실패하는 이유

좋은 모델보다 좋은 작업환경이 신뢰할 수 있는 결과를 반복해서 만든다.

실력 좋은 작업자를 아무 준비도 없는 공정에 세운다고 좋은 품질이 반복되지는 않는다. 작업 순서가 없고, 필요한 공구가 흩어져 있고, 부품 방향을 거꾸로 끼울 수 있으며, 합격 기준도 사람마다 다르다면 숙련자의 능력은 소방 활동에 소모된다.

AI도 같다. 더 큰 모델을 가져와 "알아서 잘해"라고 말하는 것으로는 운영 가능한 시스템이 만들어지지 않는다. 최근 말하는 Harness Engineering은 모델이 안정적으로 일하도록 주변 환경을 설계하는 일이다.

5.1 Harness를 현장 언어로 번역하면

여섯 개의 대응만 기억해도 충분하다. **Value Stream Mapping ↔ End-to-End Workflow Map**은 가치가 처음부터 끝까지 어떻게 흐르는지를 묻는다. **Standard Work ↔ Instruction·SOP·Skill**은 좋은 방법을 어떻게 반복할지를 묻는다.

Work Combination Table ↔ Human-Agent Work Design은 언제 누가 무엇을 수행하는지를 묻는다. **Jidoka·Andon ↔ Validation·Stop·Alert·Escalation**은 이상을 어떻게 발견하고 멈출지를 묻는다.

Poka-Yoke ↔ Permission·Schema·Rule Check는 실수를 애초에 어렵게 만드는가를 묻는다. **Control Plan·Kaizen ↔ Eval·Monitoring·Improvement**는 품질을 어떻게 측정하고 기준을 바꿀지를 묻는다.

이 대응은 번역 사전이 아니다. 개념을 일대일로 같다고 주장하지 않는다. 새로운 용어를 익숙한 운영 질문으로 바꾸는 도구다.

OpenAI는 Harness Engineering을 에이전트가 일할 환경을 설계하고, 의도를 명확히 기술하며, 신뢰할 수 있는 피드백 루프를 만드는 엔지니어링으로 설명한다. 이를 현장 언어로 풀면 다음과 같다.

Source of Truth는 어느 문서와 데이터가 기준인지 정한다. **Tooling**은 어떤 도구를 어떤 권한으로 사용할 수 있는지 정한다.

Standard Work는 목적, 순서, 완료조건, 금지행동을 명확히 한다. **Poka-Yoke**는 잘못된 입력과 위험한 실행을 구조적으로 막는다.

Control Plan은 어떤 품질을 얼마나 자주 검사할지 정한다. Andon은 문제가 생겼을 때 누가 무엇을 보고 대응할지 정한다.

5.2 Eval은 시험문제집이다

AI 평가는 막연한 만족도 조사가 아니다. 실제 업무에서 자주 발생하는 사례를 모아 반복 실행하고, 결과와 경로를 판정하는 시험 체계다.

뉴스 보고서라면 평가 세트에 다음 사례를 넣을 수 있다. 같은 사건을 다룬 기사 세 개, 정정 보도가 나온 기사, 숫자 단위가 다른 기사, 유료벽으로 원문을 볼 수 없는 기사, 회사와 관련 있어 보이지만 실제 영향은 작은 기사. 성공 조건도 구체적으로 적는다. 날짜를 맞췄는가, 원문 링크가 있는가, 사실과 의견이 분리됐는가, 근거 없는 단정이 없는가.

한 번의 멋진 데모보다 50개의 평범하고 까다로운 사례를 안정적으로 통과하는 시스템이 더 가치 있다. 실패가 발견되면 프롬프트만 고치는 것이 아니라 입력, 도구, 규칙, 권한, 평가 기준 중 어디가 원인인지 살핀다.

5.3 Loop가 PDCA가 되려면

에이전트가 계획 → 실행 → 확인 → 재시도를 반복한다고 해서 곧바로 PDCA는 아니다. 같은 실수를 그 자리에서 여러 번 고치는 것은 재작업일 수 있다. 조직의 학습이 되려면 실패가 다음 실행의 기준을 바꿔야 한다.

목표 설정 → 실행 → 결과 측정 → 원인 판단 → 수정 → 표준 · 평가 갱신

마지막 단계가 중요하다. 어떤 기사에서 날짜를 틀렸다면 그 보고서만 고치는 데서 끝내지 않는다. 기사 발행일과 사건 발생일을 별도 필드로 만들고, 혼동 사례를 평가

세트에 추가한다. 이때 루프는 Kaizen이 된다.

> **모델 교체 전에 물을 것**
>
> 실패 원인이 정보 부족인지, 도구 오류인지, 권한 문제인지, 완료조건의 모호함인지 먼저 구분하라. 환경의 결함을 더 비싼 모델로 가리면 비용만 늘고 같은 문제가 돌아온다.

Harness의 목적은 AI를 꽁꽁 묶는 것이 아니다. 중요한 판단에는 자유도를 주고, 반복되는 실수에는 구조적 장치를 두는 것이다. 좋은 치공구가 작업자의 손기술을 없애는 대신 품질 변동을 줄이듯, 좋은 Harness는 AI와 사람의 판단을 더 가치 있는 곳에 쓰게 한다.

결국 질문은 "어떤 모델이 가장 똑똑한가"에서 "이 업무가 좋은 결과를 반복하도록 무엇을 보이게 하고, 무엇을 막고, 무엇을 학습시킬 것인가"로 바뀐다. 이 전환이 데모와 운영의 경계를 만든다.

06
지식노동의 생산시스템을 설계하라

비유가 깨지는 지점을 인정하고, 조직이 내일부터 적용할 수 있는 6단계 운영모델을 제안한다.

Lean과 Agentic AI의 연결은 강력하지만, 비유를 끝까지 밀어붙이면 오히려 중요한 차이를 놓친다. 공장의 품질은 치수, 토크, 불량률처럼 비교적 명확하게 측정할 수 있다. 지식노동의 결과는 "좋은 전략인가", "충분히 설득적인가", "이 상황에서 맞는 판단인가"처럼 정답이 모호하다.

AI는 같은 입력에도 다른 답을 낼 수 있고, 틀린 결과를 매우 그럴듯하게 표현한다. 디지털 오류는 물리적 이동 없이 수천 명에게 즉시 복제된다. 사용자가 AI의 제안을 받아들이는 방식까지 시스템 성능에 영향을 준다. 그래서 제조 원리를 그대로 복사할 수는 없다.

6.1 비유가 깨지는 네 지점

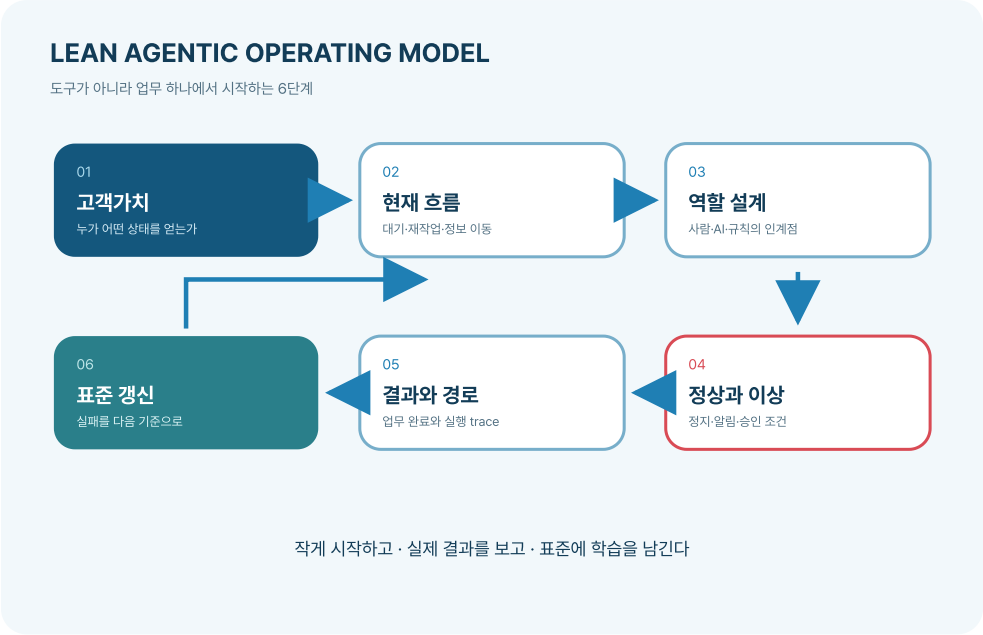
첫째는 **품질의 모호성**이다. 지식노동은 하나의 정답보다 판단 기준과 복수의 평가자가 필요하다. 둘째는 **확률적 행동**이다. 같은 조건에서도 출력이 달라질 수 있어 반복 평가와 통계적 관찰이 필요하다.

셋째는 **오류의 전파속도**다. 잘못된 문서와 결정은 복사·공유·자동실행으로 순식간에 퍼진다. 넷째는 **인간과의 상호작용**이다. 사용자의 과신, 무시, 피로가 시스템 품질을 바꾼다.

따라서 Digital Jidoka에는 공장보다 더 풍부한 평가와 모니터링이 필요하다. 단순한 합격·불합격 외에 근거성, 불확실성, 위험도, 사람의 이해 가능성을 보아야 한다. NIST가 인간-AI 피드백과 드리프트, 분산된 로그를 모니터링의 난제로 지적하는 이유다.

6.2 내일부터 적용하는 6단계

AI 도구를 고르기 전에 업무 하나를 선택해 다음 질문에 답해보자.



고객가치에서 표준 갱신까지 이어지는 Lean Agentic Operating Model · MUMULAB

1 — 고객가치를 한 문장으로 쓴다.

"뉴스를 요약한다"가 아니라 "팀장이 월요일 9시 전에 산업 변화와 우리 회사의 대응 질문을 5분 안에 이해한다"처럼 쓴다. 산출물이 아니라 사용자가 얻는 상태를 정의한다.

2 — 현재 흐름을 실제로 그린다.

요청부터 사용까지 사람, 문서, 시스템, 대기, 재작업을 표시한다. Lean Enterprise Institute가 설명하는 Value Stream Mapping처럼 부분의 효율보다 전체 흐름을 본다. 채팅창 안의 30초 절감보다 승인 대기 이틀이 더 큰 낭비일 수 있다.

3 — 사람·AI·규칙의 역할을 나눈다.

목적과 책임은 사람에게, 대량 탐색과 초안은 AI에게, 확정적 검사는 코드와 규칙에 배치한다. 누가 무엇을 잘하는가보다 인계점과 필요한 정보가 명확한지 본다.

4 — 정상과 이상을 정의한다.

완료조건, 금지조건, 허용 도구, 신뢰할 출처를 적는다. 무엇이 없으면 멈추는지, 어떤 위험이면 사람 승인이 필요한지 정한다.

5 — 결과와 경로를 함께 측정한다.

실제 업무가 끝났는지, 정확한 도구와 데이터로 끝냈는지 본다. 리드타임, 재작업률, 인간 검토시간, 오류 유출률, 예외 해결시간을 추적한다.

6 — 실패를 새 표준으로 바꾼다.

사고를 고친 뒤 같은 형태의 사례를 평가 세트에 추가한다. 지침, 권한, 데이터, 도구, 정지조건을 갱신한다. 학습이 문서와 시스템에 남아야 개선이다.

AX는 AI를 많이 쓰는 회사가 되는 일이 아니다. 인간과 AI가 함께 고객가치를 만드는 Production System을 설계하는 일이다.

6.3 사람의 역할은 사라지기보다 이동한다

Lean의 중요한 원칙은 Respect for People이다. 사람을 반복작업의 부품으로 보는 것이 아니라, 문제를 발견하고 원인을 분석하고 더 좋은 표준을 만드는 존재로 본다. AI가 Routine Execution을 가져갈수록 사람의 역할은 실행자에서 감독자, 문제해결자, 시스템 설계자로 이동할 가능성이 크다.

자동화가 늘수록 사람의 숙련이 덜 필요하다는 단순한 결론도 경계해야 한다. Toyota 역시 자동화가 진행될수록 그것을 다루고 개선할 사람의 역량이 중요하다고 강조한다. AI가 일을 대신할수록 사람은 더 적게 생각하는 것이 아니라, 더 높은 수준에서 목적과 기준과 예외를 생각해야 한다. AX는 AI를 많이 쓰는 회사가 되는 일이 아니다. 인간과 AI가 함께 고객가치를 만드는 Production System을 설계하는 일이다.

6.4 결론

AI의 미래를 이해하려면 AI 산업만 볼 필요는 없다. 인간과 기계가 함께 일하는 시스템을 가장 오랫동안 다듬어온 제조업을 다시 볼 수 있다. 작업조합, 표준작업, Jidoka, Andon, Poka-Yoke, PDCA는 완성된 답이 아니라 좋은 질문을 준다.

무엇이 가치인가. 전체 흐름은 어떻게 이어지는가. 누가 언제 일하는가. 무엇을 정상으로 볼 것인가. 언제 멈출 것인가. 실패에서 무엇을 배워 표준을 바꿀 것인가.

이 질문에 답할 때 AI 도입은 도구 구매를 넘어선다. 이메일과 메신저와 엑셀과 개인의 머릿속을 떠돌던 지식노동에 처음으로 관찰 가능한 흐름, 품질 기준, 예외관리, 지속개선의 구조가 생긴다. 그것이 Lean Manufacturing에서 Lean Agentic Enterprise로 이어지는 출발점이다.

시작은 거대한 전사 플랫폼이 아니다. 매주 반복되며, 실패 기준을 설명할 수 있고, 담당자가 결과를 직접 확인할 수 있는 업무 하나면 충분하다. 그 작은 흐름에서 사람과 AI가 함께 일하는 표준을 만들고, 실제 결과로 다음 표준을 갱신하라.

6.5 참고자료

- Toyota Motor Corporation, **Toyota Production System**: https://global.toyota/en/company/vision-and-philosophy/production-system/

- Lean Enterprise Institute, **Standardized Work**: https://www.lean.org/lexicon-terms/standardized-work/

- Lean Enterprise Institute, **Value-Stream Mapping**: https://www.lean.org/lexicon-terms/value-stream-mapping/

- OpenAI, **Harness engineering: leveraging Codex in an agent-first world**: https://openai.com/index/harness-engineering/
- Anthropic, **Building effective agents**: https://www.anthropic.com/engineering/building-effective-agents
- Anthropic, **Demystifying evals for AI agents**: https://www.anthropic.com/engineering/demystifying-evals-for-ai-agents
- Microsoft Learn, **Responsible AI for agentic systems**: https://learn.microsoft.com/en-us/agents/center-of-excellence/responsible-ai
- NIST, **Challenges in Monitoring Deployed AI Systems**: https://www.nist.gov/news-events/news/2026/03/new-report-challenges-monitoring-deployed-ai-systems