PLATFORM ENGINEERING PARA LA EMPRESA AI-NATIVE · PLAYBOOK

La plataforma, no el modelo, decide si la IA empresarial entrega.

Tres cosas son verdaderas a la vez en 2026: la adopción de IA es casi universal, la mayoría de los proyectos de IA todavía fracasa en producción, y los que tienen éxito comparten una plataforma madura por debajo. Este playbook sostiene que la fundación decide, mapea la pirámide de cinco capas y el plano de control de cuatro pilares, entrega dos aceleradores sobre una fundación Azure y GitHub, y convierte la estrategia en una secuencia de 90/180/365 días medida en DORA y GitHub AI Credits.

AUTORA

Paula Silva

CARGO

AI-Native Software Engineer

CONTACTO

in/paulanunes

TIEMPO DE LECTURA

~ 45 minutos, de punta a punta

5

CAPAS DE LA PIRÁMIDE

6

CAPÍTULOS

4

PILARES DE LA CNCF

2.5x

COMPRESIÓN DE ROI,
PLATAFORMA MADURA

CAPÍTULOS

Chapter 00

Por qué la fundación decide

Los tres hechos de 2026, la tesis de la amplificación de DORA 2025, los cinco modos de fallo cuando la plataforma es débil, y la evidencia que un CTO lleva al directorio.

El Argumento →

Chapter 01

La Pirámide de Cinco Capas

Plataforma, Contexto, Cognición, Intención y Agentic en estricto orden de dependencia, la regla de apilamiento que gobierna la construcción, y dónde encaja GitHub Copilot en cada capa.

Arquitectura →

Chapter 02

Los Cuatro Pilares como Plano de Control

Golden Paths, Guardrails, Safety Nets y Manual Review reexpresados para gobernar agentes con la maquinaria que la plataforma ya opera, vistos a través de la lente de Copilot y del Doble Mandato de 2026.

Plano de Control →

Chapter 03

Dos Aceleradores, Una Fundación

Three Horizons sobre Red Hat Developer Hub y Open Horizons sobre Backstage, la fundación Azure y GitHub que no cambia, y un método de un día para elegir por el arquetipo del cliente.

La Elección →

Chapter 04

La secuencia de 90/180/365 días

El plan de ejecución por fases de la infraestructura Azure bruta a los agentes en producción en un año, y los tres innegociables que deciden si la secuencia se concreta.

Ejecución →

Chapter 05

Madurez, medición y ROI

El modelo de madurez de la CNCF en cinco dimensiones, las Cuatro Métricas Clave DORA más las extensiones de IA de 2026 y la salud de agentes, gobernanza de costo en GitHub AI Credits, y un business case defendible a nivel del CFO.

El Retorno →

CONSEJOS DE LECTURA

> **ATAJOS DE TECLADO**
>
> Presiona `/` para buscar. Usa `j` y `k` para moverte entre capítulos.
> Presiona `t` para cambiar tema. Presiona `g` para ir al inicio.

Paula Silva

AI-NATIVE SOFTWARE ENGINEER

in/paulanunes

v1.0.0 · 2026-07-02

Building the future of software development with AI and Agentic DevOps

00 EL ARGUMENTO

Por qué la fundación decide: la plataforma, no el modelo, fija el techo.

Tres cosas son verdaderas a la vez en 2026. La adopción de IA es casi universal. La mayoría de los proyectos de IA todavía fracasa en producción. Y las organizaciones que tienen éxito comparten un rasgo: una plataforma madura por debajo. Este capítulo sostiene que la fundación, no el modelo, decide el resultado. Lee la evidencia de 2025 y 2026 como un CTO tiene que leerla: como una decisión de gasto con un riesgo medible.

Los tres hechos de 2026

Empieza por la adopción. El informe The State of AI 2025 de McKinsey encuentra que el 78% de las organizaciones ya usa IA en al menos una función. En las empresas con las que trabajo por América Latina, esa adopción suele tener un nombre: GitHub Copilot, y cada vez más el Copilot en agent mode corriendo por todo el parque de repositorios. Gartner proyecta que el 40% de las aplicaciones empresariales incorporará agentes para tareas específicas para fines de 2026, frente a menos del 5% en 2024. La adopción está resuelta. La pregunta abierta no es si los equipos usan IA. Es qué le hace esa IA al sistema de entrega sobre el que aterriza.

El segundo hecho es menos cómodo. El estudio The GenAI Divide, del MIT NANDA, publicado en 2025 con más de 350 empresas, encontró que el 95% de los pilotos de IA generativa no entrega valor medible. El fracaso allí se define de forma operativa: el piloto nunca llegó a producción sostenida con resultados de negocio atribuibles. El tercer hecho reconcilia los dos primeros. Las organizaciones que sí llegan a producción son, con consistencia notable, las maduras en plataforma. NANDA atribuye la diferencia no a modelos débiles sino a la ausencia de contexto a nivel de sistema y de bucles de feedback. Eso es una propiedad de la plataforma, no del

modelo. La misma herramienta tiene éxito en un parque y fracasa en el siguiente, y lo que cambió es la fundación por debajo.

La tesis de la amplificación

El mecanismo detrás de esos tres hechos tiene nombre, y la evidencia más fuerte es DORA. El informe DORA 2025 se apoya en casi 5.000 respuestas de encuesta y más de 100 horas de investigación cualitativa. Su hallazgo central es que la IA no se comporta como un insumo universal de productividad. Se comporta como amplificadora del sistema de entrega que hereda. Los equipos en arquitecturas débilmente acopladas con feedback rápido convierten las mismas herramientas de IA en throughput real. Los equipos en sistemas fuertemente acoplados con feedback lento ven el volumen extra de cambios convertirse en inestabilidad. El mismo insumo, resultados opuestos. DORA también reporta que el 90% de las organizaciones encuestadas adoptó al menos una plataforma interna, lo que mueve la conversación de si construir una plataforma a qué tan madura es la tuya.

Aquí es donde importa la lente de GitHub Copilot. Cuando una empresa enciende el agent mode en miles de repositorios, amplifica el sistema de entrega que esos repositorios ya codifican. El trabajo de la plataforma es alimentar al agente con contexto autoritativo para que la amplificación vaya en la dirección correcta. En la práctica eso significa un .github/copilot-instructions.md mantenido en cada repositorio, servidores MCP que exponen el catálogo de servicios, las políticas y los runbooks, y custom agents con alcance limitado a tareas sancionadas. También hace visible el costo. El uso de Copilot se factura como GitHub AI Credits, donde un crédito equivale a un centavo de dólar, y desde junio de 2026 esa línea muestra exactamente cuánta reconstrucción de contexto está pagando el parque. Una fundación débil hace crecer ese número. Una madura lo convierte en throughput.

> DEFINICIÓN
>
> DORA 2025 lo resume en una línea: la IA no arregla un equipo; amplifica lo que ya está ahí. Léelo como una regla de gasto. Invertir en IA sin invertir en plataforma es invertir en amplificar lo que ya existe, en la dirección hacia la que ya apunta.

Cinco modos de fallo cuando la fundación es débil

Cuando la plataforma está subconstruida frente a la IA escalada encima, cinco modos de fallo se repiten. El primero es la deuda triple. La IA genera deuda técnica mediante código sin revisión, deuda cognitiva mediante conocimiento que solo existe en prompts e hilos de chat, y deuda de intención mediante objetivos implícitos o contradictorios. Las tres se acumulan en paralelo y ninguna resuelve a las otras. El segundo son las plataformas sombra. Cuando la plataforma central es difícil de usar, cada equipo construye la suya: pipelines a medida, módulos de infraestructura conflictivos, sus propios servidores MCP y custom agents. Los datos de shadow IT de Gartner sitúan ese gasto distribuido en dos a tres veces el costo de una plataforma real, invisible en cualquier línea presupuestaria.

El tercero es la pudrición de contexto. Los catálogos de servicios, la documentación y el linaje quedan obsoletos en silencio, y un agente que razona sobre un copilot-instructions.md desactualizado o una fuente MCP obsoleta devuelve respuestas plausibles y erróneas. Al volumen de los agentes, una pequeña tasa de pudrición se vuelve una gran tasa de error. El cuarto es la regresión de seguridad. El Informe Global de Amenazas 2026 de CrowdStrike documenta que las organizaciones que desplegaron asistentes de codificación con IA antes de madurar su postura de seguridad de plataforma vieron un aumento del 38% en vulnerabilidades explotables en los primeros doce meses. La regresión desaparece donde la plataforma aplica controles de cadena de suministro y de identidad en la admisión. El quinto es el problema del 100:1. Para 2028, la proporción agente-humano en una empresa típica se proyecta que llegue a cerca de 100 a 1. La supervisión manual de identidad, credenciales y auditoría de los agentes no escala a esa proporción. Solo una plataforma que trata a los agentes como cargas de trabajo de primera clase lo hace.

> DEFINICIÓN
>
> Cinco modos de fallo, una causa: la IA corre más rápido de lo que el proceso ad hoc puede gobernar. Deuda triple. Plataformas sombra a dos o tres veces el costo. Pudrición de contexto. Regresión de seguridad. La proporción de 100:1 para 2028. Cada uno es evitable por la plataforma. Ninguno es evitable por un modelo mejor.

La evidencia que un CTO lleva al directorio

Para un ejecutivo, los números decisivos son los que separan la intención de la capacidad. La investigación de IDC sobre IA agéntica encuentra que, aunque cerca de la mitad de las organizaciones encuestadas reporta diez o más agentes en alguna forma de producción, menos del 7% ejecuta siquiera un caso de uso en producción plena. La voluntad está ampliamente distribuida; la capacidad está concentrada en las organizaciones que construyeron la plataforma primero. Esa única brecha es más útil que cualquier titular de productividad, porque es observable dentro de tu propio parque hoy. Cuenta cuántos casos de uso de agentes están de verdad en producción plena, no en piloto. La respuesta te dice dónde está tu fundación.

El lado positivo también está documentado. La investigación de Platform Engineering de Forrester vincula las plataformas internas productizadas directamente con resultados de negocio: el 77% de las empresas atribuye ganancias medibles de time-to-market a su plataforma interna de desarrollo, y el 85% reporta impacto positivo en ingresos. Junta los dos hallazgos y el argumento se cierra. La restricción que limita la IA empresarial no es qué modelo está en el selector de Copilot. Claude Fable 5, Opus 4.8, Sonnet 5, Haiku 4.5, Gemini 3.1 Pro y la línea GPT-5.x están todos a un clic en el menú, y cualquiera de ellos amplificará lo que hereda. La restricción es la plataforma que los alimenta con contexto, gobierna sus acciones y paga por sus tokens. Por eso la fundación decide. Constrúyela primero.

Referencias

Fuentes primarias para los tres hechos, la tesis de la amplificación, los modos de fallo y la evidencia ejecutiva.

- McKinsey, The State of AI 2025, la línea base de adopción: el 78% de las organizaciones ya usa IA en al menos una función.
- MIT NANDA, The GenAI Divide: State of AI in Business 2025, la tasa de fallo del 95% de los pilotos y su diagnóstico como un problema de contexto y feedback.
- DORA 2025, Accelerate State of DevOps Report, el hallazgo de la amplificación, apoyado en casi 5.000 respuestas y más de 100 horas de investigación cualitativa.

- Gartner, Over 40% of Agentic AI Projects Will Be Canceled by End of 2027, la proyección de cancelaciones impulsada por costos, valor poco claro y controles de riesgo débiles.
- IDC, Agentic AI Platforms and Strategies, la brecha de preparación entre agentes declarados en producción y casos de uso de verdad en producción plena.
- Forrester Wave, Platform Engineering Solutions Q1 2026, plataformas internas de desarrollo vinculadas a time-to-market e impacto en ingresos.
- CNCF Platforms Whitepaper, la definición de referencia de las capacidades de plataforma que este capítulo defiende.
- CrowdStrike, 2026 Global Threat Report, la regresión de seguridad medida cuando los asistentes de codificación con IA llegan antes que los controles de plataforma.

Paula Silva

AI-NATIVE SOFTWARE ENGINEER

in/paulanunes

v1.0.0 · 2026-07-02

Building the future of software development with AI and Agentic DevOps

01 LA ARQUITECTURA

La Pirámide de Cinco Capas: cinco capas en estricto orden de dependencia, de la plataforma a los agentes.

La empresa AI-native no es una sola cosa. Es una pirámide de cinco capas, cada una dependiendo de la que está debajo. Platform Engineering es la base. Contexto, Cognición, Intención y Agentic se apilan encima, en ese orden. El orden no es un consejo, es una ley estructural. Este capítulo mapea las cinco capas, nombra lo que cada una posee, muestra dónde encaja GitHub Copilot y explica por qué construir agentes sobre una fundación ausente produce un desorden amplificado en lugar de agentes.

La regla de apilamiento: cada capa está en estricto orden de dependencia

La empresa AI-native es una pirámide, no una torta de capas. La capa de la base es más ancha, cuesta más y sostiene todo lo que está encima. Cada capa superior es más estrecha y explícitamente condicionada a la capa de abajo. El orden de construcción no es una preferencia, es una ley estructural: no se puede construir la capa N+1 más rápido que la capa N. Se pueden construir en paralelo, pero solo si se acepta la deuda que viene de la brecha. Una pirámide se ensambla de abajo hacia arriba. No se puede suspender la cima en el aire y construir la base a su alrededor después. La capa de Plataforma se construye primero porque toda capa por encima necesita los primitivos de plataforma para existir. Quita una piedra y no obtienes una pirámide más pequeña, obtienes un colapso.

La regla tiene base empírica. El DORA 2025, construido sobre casi 5.000 respuestas de encuesta y más de 100 horas de datos cualitativos, encontró que el 90% de las organizaciones adoptaron al menos una plataforma interna, con correlación directa entre la calidad de la plataforma y lo que la IA entrega. El informe resume el

mecanismo en una línea: "AI doesn't fix a team; it amplifies what's already there". Es decir, la IA no arregla un equipo, amplifica lo que ya existe. Esa frase es la pirámide reformulada. La IA consume la pila de abajo hacia arriba. Una base débil produce una cima débil, por más fuerte que la cima parezca de forma aislada. Las mismas herramientas de IA, resultados opuestos, y la plataforma es la diferencia.

L1 Plataforma a L5 Agentic: qué hace cada capa y qué posee cada una

La Capa 1, Platform Engineering, es el sustrato autoservicio, gobernado por políticas y observable. Posee los cuatro pilares de la CNCF: Golden Paths, Guardrails, Safety Nets y Manual Review. Los Golden Paths llevan a un desarrollador o a un agente desde "quiero construir X" hasta un sistema en cumplimiento y en ejecución, sin abrir un ticket. Los Guardrails son preventivos, aplicados en la admisión en lugar de detectados después del despliegue. Las Safety Nets se recuperan de los fallos que los guardrails no previnieron. La Manual Review mantiene a un humano en el bucle para acciones irreversibles. Los artefactos concretos son un portal Backstage, GitOps, policy-as-code, observabilidad con OpenTelemetry y workload identity. La Capa 2, Contexto, es el conocimiento empresarial codificado en formato consumible por máquinas: el catálogo de servicios, documentación como código, linaje, almacenes vectoriales y servidores MCP. Depende de la L1 porque la plataforma posee los primitivos de identidad, observabilidad y ciclo de vida que hacen que el contexto sea confiable.

La Capa 3, Cognición, son los modelos, embeddings y evaluación que consumen contexto para producir salidas, mediados por un gateway de modelos que la plataforma provisiona y gobierna. La Capa 4, Intención, son los objetivos, especificaciones y políticas codificados contra los cuales los agentes planifican. Intención sin contexto es alucinación; contexto sin intención es descripción sin dirección. La Capa 5, Agentic, son agentes autónomos y orientados a objetivos que combinan intención, contexto y cognición para actuar. Esta es la capa visible para la mayor parte de la inversión en IA en 2026, y es la capa con mayor probabilidad de fallar cuando las capas subyacentes están ausentes. Un agente es un modelo más un harness. El modelo es la parte enchufable. El harness es todo lo que lo rodea: herramientas, memoria, contexto, validación y gates. El harness son las cuatro capas

inferiores en forma operativa, y por eso un agente en producción es la prueba integrada de si la fundación es sólida.

La lente de Copilot: dónde se mapea GitHub Copilot en la pirámide

GitHub Copilot hace concreta la pirámide. La capa de contexto es donde viven los servidores MCP y el archivo .github/copilot-instructions.md. Los servidores MCP exponen sistemas empresariales (Azure, GitHub, monitoreo, tickets) a Copilot mediante un único protocolo estándar, de modo que el agente lee la realidad de la empresa en lugar de adivinarla. El archivo copilot-instructions es contexto en el alcance del repositorio: las convenciones, la arquitectura y las reglas que la base de código ya codifica, escritas donde cada sesión de Copilot las lee. Ambos son artefactos de la Capa 2. Ninguno es un modelo. Son el conocimiento sobre el cual el modelo razona, y son lo que impide que un modelo capaz produzca una salida plausible que no corresponde a cómo funciona realmente la base de código.

La capa cognitiva es el selector de modelos de Copilot. En 2026 el selector ofrece Claude Fable 5, Opus 4.8, Sonnet 5, Haiku 4.5, Gemini 3.1 Pro y la familia GPT-5.x. Ningún modelo es el mejor para toda tarea, así que el selector es el gateway de modelos hecho visible: diriges un modelo rápido y barato a una edición acotada y un modelo de razonamiento más fuerte a un refactor difícil. El uso se factura en GitHub AI Credits, donde un crédito equivale a un centavo de dólar, en la medición que entró en vigor el 1 de junio de 2026. El agent mode de Copilot y los agentes personalizados son la superficie de la Capa 5: combinan las instrucciones (intención y contexto), el modelo elegido (cognición) y los controles de la plataforma para actuar sobre el repositorio. El selector sin las instrucciones y sin MCP es cognición sin anclaje.

Saltar a L5 sin L1 a L4 no produce agentes

El fallo más común es financiar la capa Agentic, que es visible y fácil de financiar, sin financiar las capas de Plataforma y Contexto, que son invisibles y sin glamour. El resultado es un agente que funciona en una demo y falla en producción. No tiene contexto anclado, así que alucina. No tiene permisos gobernados, así que actúa

fuera de su alcance. No tiene trayectorias observables, así que nadie puede depurar ni certificar lo que hizo. Gartner estima que el 40% de los proyectos de IA agéntica serán cancelados antes de 2027 por costo, valor poco claro y controles de riesgo débiles. Estructuralmente, esos son despliegues de agentes sobre fundaciones inadecuadas: una pirámide a la que le faltan una o más capas inferiores.

Una organización que salta a L5 sin L1 a L4 en su lugar no produce agentes. Produce un desorden amplificado. La corrección nunca es arreglar el agente. Es reajustar las capas ausentes, en orden de dependencia, lo que cuesta más después de que el agente ya fue mostrado a ejecutivos de lo que habría costado construir bien la primera vez. La versión Copilot del mismo error es entregar a un equipo el modelo más fuerte del selector sin ningún archivo copilot-instructions y sin ningún servidor MCP conectado. El modelo no es la restricción. El contexto y la plataforma debajo de él lo son. Empieza en L1. Comprométete con la fundación antes que con los agentes.

> DEFINICIÓN
>
> La regla de apilamiento en una línea: no se puede construir la capa N+1 más rápido que la capa N, y solo se pueden construir en paralelo si se está dispuesto a cargar la deuda que viene de la brecha. Saltar a L5 sin L1 a L4 no produce agentes. Produce un desorden amplificado.

Referencias

Fuentes primarias para la pirámide de cinco capas, la regla de apilamiento y las fundaciones de plataforma y contexto sobre las que se apoya.

- CNCF Platforms Whitepaper, la definición de referencia de la capa de plataforma como un sustrato autoservicio, gobernado por políticas y observable.
- CNCF Platform Engineering Maturity Model, la fuente de los cuatro pilares que posee la Capa 1: Golden Paths, Guardrails, Safety Nets y Manual Review.
- DORA 2025, State of AI-assisted Software Development, la base empírica de la amplificación: la IA amplifica lo que ya existe.
- Backstage, el portal de desarrollador open-source que materializa los Golden Paths y el catálogo de servicios en la base de la pirámide.

- Model Context Protocol Specification, el estándar abierto que los servidores MCP usan para exponer sistemas empresariales como contexto consumible por agentes.
- Vishnyakova et al., From Context Engineering to Multi-Agent Architecture, la jerarquía de dependencia de contexto a intención que las capas superiores de la pirámide formalizan.

Paula Silva

AI-NATIVE SOFTWARE ENGINEER

in/paulanunes

v1.0.0 · 2026-07-02

Building the future of software development with AI and Agentic DevOps

02 EL PLANO DE CONTROL

Los Cuatro Pilares como Plano de Control: cómo una plataforma gobierna agentes con la maquinaria que ya opera.

Golden Paths, Guardrails, Safety Nets y Manual Review fueron diseñados para gobernar la entrega humana. En 2026 los mismos cuatro pilares gobiernan agentes. La reexpresión es operativa, no metafórica. Una plataforma que codificó los cuatro pilares para desarrolladores ya posee la mayor parte del plano de control de IA. Este capítulo recorre los cuatro pilares como la superficie que admite, contiene, recupera y aprueba cada acción que ejecuta un agente.

Golden Paths y Guardrails: la superficie sancionada y el campo de contención

Un agente es un servicio. Esa única equivalencia es lo que permite a una plataforma gobernar agentes con la maquinaria que ya opera. Un Golden Path es un flujo de trabajo autoservicio y con criterio que lleva a un desarrollador, o a un agente, desde "quiero construir X" hasta un sistema en cumplimiento y funcionando en minutos. Para los agentes, el Golden Path se convierte en el scaffold mediante el cual se instancia el agente. La regla es directa: si no es un Golden Path, el agente no lo ejecuta. El Golden Path es la superficie de ejecución sancionada, y todo lo que queda fuera de él está fuera de la superficie.

Los Guardrails son la otra mitad. Son preventivos, no detectivos. Un control detectivo señala una violación después del despliegue, lo que es demasiado lento cuando los agentes actúan más rápido que cualquier bucle de revisión humana. Un control preventivo rechaza la violación en la admisión, antes de que aterrice. La misma política de OPA Gatekeeper o Kyverno que rechaza un manifiesto Kubernetes no conforme rechaza una configuración de agente no conforme. El mismo Workload

Identity que emite credenciales de corta duración para un servicio las emite para un agente. La acción incorrecta no recibe una advertencia. Se vuelve imposible.

El campo de contención del agente

Sobre los guardrails compartidos, la plataforma agrega cuatro que son específicos de agentes. Un guardrail de alcance de permisos declara los datos que un agente puede leer y las acciones que puede tomar, aplicado en tiempo de ejecución. Un guardrail de filtro de salida redacta y clasifica antes de que la salida llegue al sistema consumidor. Un guardrail de residencia de datos enruta el tráfico solo a endpoints de modelo en regiones permitidas. Un guardrail de circuit breaker de costos termina una sesión cuando el gasto cruza un techo. Juntos, esos cuatro forman el campo de contención del agente, aplicado en la admisión y de nuevo en tiempo de ejecución.

> DEFINICIÓN
>
> Los Golden Paths son la única manera sancionada de que un agente arranque. Los Guardrails son lo que debe satisfacer para seguir corriendo. Uno admite, el otro contiene. Ninguno es opcional.

Safety Nets y Manual Review: recupera lo reversible, gatea lo irreversible

Los Guardrails previenen. Los Safety Nets recuperan de lo que la prevención dejó pasar. El Safety Net más fundamental es la reconciliación GitOps: Argo CD o Flux comparan continuamente el estado desplegado con el estado declarado en Git y corrigen la desviación sin humano en el bucle. Reexpresado para agentes, el mismo bucle se convierte en rollback a nivel de decisión. Cuando la salida de un agente se desvía de su especificación declarada, cuando el costo por tarea sube, o cuando la tasa de éxito cae, el Safety Net revierte la decisión o escala. La detección de bucles aborta una sesión que se repite sin progreso. Un congelamiento orientado por SLO detiene una población de agentes en el instante en que se viola un objetivo de servicio, del mismo modo que un canary detiene un rollout defectuoso.

Manual Review es el último pilar y el que gana importancia a medida que todo lo demás se acelera. Su función es insertar juicio humano en las fronteras que lo justifican, y en ninguna otra. Los cambios de rutina fluyen por Guardrails y Safety Nets sin toque. Las acciones irreversibles chocan con un gate de aprobación: una migración de datos en producción, una expansión de alcance de permisos, un gasto por encima de un límite. Para los agentes, el gate crítico es la expansión de capacidad. Antes de que un agente gane una nueva fuente de datos o una nueva acción, un humano lo aprueba. Ese es el equivalente en IA de un despliegue elevado en producción, y es el único gate que una plataforma madura nunca automatiza hacia afuera.

El agente auditor

La fricción es intencional, pero la recolección de evidencia no tiene por qué ser manual. Un agente auditor lee el cambio propuesto, recupera la entrada del catálogo de servicios, el historial de despliegues e incidentes, el impacto de costos y el estado de cumplimiento de políticas, y luego arma un paquete de revisión estructurado. El humano gasta tiempo en la decisión, no en reunir los hechos. La carga cognitiva sobre el revisor cae. La responsabilidad permanece exactamente donde estaba. Este es un ejemplo limpio de IA aumentando los cuatro pilares en lugar de reemplazarlos.

La lente de Copilot: guardrails que un agente satisface en tiempo de ejecución

Los cuatro pilares siguen siendo abstractos hasta que tocan una herramienta que el desarrollador ya usa. GitHub Copilot en agent mode es donde los aterrizo. Un archivo .github/copilot-instructions.md de un repositorio es un guardrail escrito como texto. Declara las convenciones, las fronteras y los patrones sancionados que el agente debe honrar en cada tarea de ese repositorio. Los custom agents estrechan aún más la superficie: cada uno está acotado a un dominio, lleva sus propias instrucciones y solo alcanza las herramientas y el contexto que le concedes. Ese acotamiento es un guardrail de alcance de permisos expresado en la capa de Copilot, y no en la capa del clúster.

El MCP es cómo el agente alcanza contexto y herramientas bajo gobernanza. Un servidor MCP expone un conjunto definido de recursos y acciones, y la plataforma decide qué agentes se conectan a qué servidores. El modelo en sí viene del selector de Copilot, y el selector es su propio tipo de guardrail: un agente solo puede correr en un modelo que la organización habilitó, lo que hoy significa Claude Fable 5, Claude Opus 4.8, Claude Sonnet 5, Claude Haiku 4.5, Gemini 3.1 Pro o un modelo GPT-5.x. Desde el 1 de junio de 2026, el uso premium se cobra en GitHub AI Credits, donde un crédito equivale a un centavo de dólar. Esa superficie de cobro es el guardrail de circuit breaker de costos en la práctica: el gasto se mide por solicitud, es atribuible por equipo y se limita antes de descontrolarse.

> DEFINICIÓN
>
> En la lente de Copilot, el .github/copilot-instructions.md es el guardrail, el custom agent es el alcance de permisos, el MCP es la frontera de contexto gobernada, y los GitHub AI Credits son el circuit breaker de costos. Los cuatro pilares no son un diagrama. Son configuración.

El Doble Mandato de 2026

Los cuatro pilares describen cómo una plataforma gobierna agentes. El Doble Mandato describe lo que un equipo de plataforma le debe a la empresa mientras lo hace. El Mandato A es aumentar la propia plataforma con IA para velocidad interna. Los agentes hacen triaje de alertas, redactan correcciones de runbook, proponen cambios de infraestructura, revisan policy-as-code y generan invocaciones de Golden Path en el IDE. El objetivo es que las mejoras de plataforma se entreguen más rápido, y cada mejora que aterriza antes se compone en todos los equipos que consumen la plataforma. La disciplina que mantiene honesto al Mandato A es la adopción medida. Si menos de un tercio de las acciones diarias del equipo realmente pasa por los agentes, las herramientas son teatro, no integración.

El Mandato B es exponer la IA como primitivos de plataforma que los equipos de aplicación consumen a través de Golden Paths para velocidad externa. La inferencia por un gateway de modelos gobernado, el almacenamiento vectorial, el runtime de agentes, la evaluación y la observabilidad de agentes se convierten en superficies

de primera clase, del mismo modo que Kubernetes y Postgres se volvieron de primera clase una década antes. Azure AI Foundry es el gateway de referencia en ambos aceleradores. El objetivo es que los equipos de aplicación entreguen productos que usan IA en semanas, y no en trimestres. Una plataforma que ejecuta solo el Mandato A produce un equipo de plataforma más rápido que no mueve a la empresa. Una plataforma que ejecuta solo el Mandato B se convierte en un buffet de IA sin velocidad propia, y al final en el cuello de botella. Las plataformas maduras ejecutan los dos.

> DEFINICIÓN
>
> La prueba de madurez es una pregunta en dos mitades. ¿Puedes nombrar una tarea recurrente del equipo de plataforma que un agente haya absorbido, y un equipo de aplicación cuyo cada primitivo de IA es un Golden Path de tu plataforma? Dos síes es una plataforma madura. Un sí es media plataforma. Ninguna de las respuestas es opcional en 2026.

Referencias

Fuentes primarias para los cuatro pilares, la reexpresión del plano de control de agentes y la superficie de ejecución de Copilot.

- CNCF Platform Engineering Maturity Model, la taxonomía de referencia para Golden Paths, Guardrails, Safety Nets y Manual Review, y los niveles de madurez por pilar.
- Backstage Software Templates (Scaffolder), el único punto de entrada de cara al desarrollador que convierte un Golden Path en la superficie de ejecución sancionada.
- Open Policy Agent Gatekeeper, aplicación de política en el momento de la admisión, el mecanismo que convierte un guardrail en un error de compilación para la infraestructura.
- Argo CD Documentation, reconciliación GitOps, el Safety Net fundamental que corrige la desviación de vuelta al estado declarado.

- GitHub Copilot Documentation, la superficie de ejecución de referencia para agent mode, custom agents y .github/copilot-instructions.md.
- Microsoft Entra Agent ID, identidad gestionada de agente, el primitivo que permite a los mismos guardrails gobernar agentes y servicios de forma uniforme.
- Azure AI Foundry Documentation, el gateway de modelos gobernado en el centro del Mandato B.
- DORA Report 2025, el hallazgo de la amplificación: la IA no arregla un equipo, amplifica lo que la plataforma ya es.

Paula Silva

AI-NATIVE SOFTWARE ENGINEER

in/paulanunes

v1.0.0 · 2026-07-02

Building the future of software development with AI and Agentic DevOps

03 LA ELECCIÓN

Dos Aceleradores, Una Fundación.

Los dos aceleradores llegan al mismo lugar. Three Horizons y Open Horizons construyen la misma plataforma AI-Native, sobre la misma fundación Azure y GitHub, y ambos terminan en el mismo Horizon 3. Lo que difiere es el stack en el medio: si la internal developer platform corre sobre Red Hat Developer Hub con soporte comercial, o sobre Backstage que tu propio equipo opera. Este capítulo presenta los dos, nombra la fundación que no se mueve y entrega un método de un día para elegir.

Three Horizons: Red Hat Developer Hub sobre Azure Red Hat OpenShift

Three Horizons es el acelerador para clientes con una inversión previa en Red Hat o un requisito de soporte comercial. La internal developer platform es Red Hat Developer Hub. El clúster es Azure Red Hat OpenShift, con el control plane operado en conjunto por Microsoft y Red Hat bajo un SLA conjunto. El CI corre sobre OpenShift Pipelines, el GitOps sobre OpenShift GitOps, y Red Hat mantiene soporte enterprise las 24 horas. Los sectores regulados que exigen que cada componente de la plataforma lleve su propia certificación, FedRAMP, FIPS o Common Criteria, suelen aterrizar aquí, porque los componentes del stack de Red Hat poseen esas certificaciones y los equivalentes open-source no.

El paquete de suscripciones de Red Hat suele representar entre el 10 y el 30 por ciento del gasto total de plataforma, escalando con el tamaño del clúster, la cantidad de usuarios y la cobertura de add-ons. Ese ítem de costo compra un modelo de soporte. El requisito interno del equipo de plataforma se mantiene operativo, ejecutar y personalizar la plataforma, en lugar de ingeniería profunda, porque los problemas del stack upstream se derivan a Red Hat y no a tus propios desarrolladores. Para un banco bajo las reglas del BACEN o una operadora de energía con una flota RHEL a escala, Red Hat Developer Hub sobre OpenShift es continuidad, no costo extra.

Open Horizons: Backstage sobre Azure a través de los tres horizontes

Open Horizons es un único repositorio template en GitHub que aprovisiona la misma plataforma en Azure usando componentes open-source de la CNCF. El contenido es concreto: 16 módulos Terraform para la fundación Azure Kubernetes Service, 22 plantillas de Golden Path distribuidas entre los tres horizontes, 13 configuraciones de servidor MCP y 17 agentes personalizados de GitHub Copilot conectados para el Doble Mandato. Backstage es la internal developer platform, y tu propio equipo la opera. Un equipo que adopta Open Horizons no escribe una plataforma desde cero. Personaliza un punto de partida con criterio, lo que representa una diferencia de un orden de magnitud en tiempo, costo y riesgo.

La lente de GitHub Copilot es explícita aquí. Los 17 agentes personalizados corren en agent mode; las convenciones que sigue cada agente viven en .github/copilot-instructions.md; y las 13 configuraciones de servidor MCP dan a cada agente acceso tipado y acotado por rol a Azure, GitHub, Terraform, Kubernetes y el stack de observabilidad. El consumo de los agentes se factura como GitHub AI Credits, donde un crédito equivale a un centavo de dólar. El intercambio frente a Three Horizons es directo. No hay ítem de suscripción de Red Hat, y el requisito de capacidad interna es más amplio, porque los problemas upstream se derivan a la comunidad y a tus propios ingenieros, no a un SLA de proveedor.

La fundación que no cambia

Bajo ambos aceleradores el sustrato es idéntico, y vale la pena nombrarlo con precisión justamente porque es lo que no se mueve. Azure es la nube. GitHub es la fuente de registro y la superficie de CI. Un único plano de identidad pasa por Microsoft Entra, de modo que personas y agentes se gobiernan en el mismo lugar. Un único model picker de GitHub Copilot expone los mismos modelos a cada desarrollador y a cada agente, sin importar el acelerador que esté encima, incluidos Claude Opus 4.8 y Sonnet 5, Gemini 3.1 Pro, la línea GPT-5.x y opciones más livianas como Haiku 4.5 para trabajo barato y de alto volumen. Los cuatro pilares de la CNCF, el modelo de madurez H1, H2 y H3, y el Doble Mandato son los mismos en ambos.

Por eso la decisión es una elección de stack, no una elección de estrategia. La estrategia no varía: la misma pirámide de cinco capas, los mismos cuatro pilares, el mismo Doble Mandato, el mismo camino de 90 a 180 días hasta el Horizon 3. Lo que varía es el medio del stack, la internal developer platform, el clúster, el sustrato de CI, las herramientas de cadena de suministro y si las suscripciones de Red Hat están en el contrato. Ambos caminos terminan en el mismo destino Azure AI Foundry, exponiendo los mismos Golden Paths, los mismos servidores MCP y los mismos agentes.

> DEFINICIÓN
>
> Mismo destino, misma fundación, stack distinto por encima. Azure para la nube, GitHub para código y CI, un único plano de identidad Microsoft Entra para personas y agentes, un único model picker de GitHub Copilot para ambos. El acelerador que eliges cambia el medio del stack, nunca la estrategia y nunca el punto final.

Elegir por el arquetipo del cliente

La mayoría de los clientes se inclina antes de analizar, y la inclinación suele ser correcta. Un retailer digital-native ya sobre Azure Kubernetes Service, con una cultura OSS-first y una cantidad de proveedores deliberadamente pequeña, se inclina hacia Open Horizons. Un banco regulado por el BACEN con un parque OpenShift on-premise y una cláusula de soporte comercial en el procurement se inclina hacia Three Horizons. La inclinación viene del perfil de procurement, la capacidad de ingeniería, la postura regulatoria y la estrategia de proveedores, no de una lista de características. Cerca de un cuarto a un tercio de las empresas queda en un grupo intermedio donde ninguna inclinación es dominante, y ahí es donde un método importa.

El descubrimiento de un día

Para el grupo intermedio conduzco un taller de descubrimiento de un día. Dos semanas antes, el cliente envía un inventario: el parque Red Hat actual, el parque Kubernetes actual, la inversión en GitHub, el perfil del equipo de ingeniería y la

postura de procurement y regulatoria. El día tiene cinco sesiones de trabajo: alineación de estrategia, un recorrido por la comparación de stack en diez dimensiones, un modelo de costo total de propiedad a tres años construido con los números del propio cliente, una autoevaluación honesta de la capacidad del equipo para operar Backstage y Azure Kubernetes Service upstream, y una recomendación. La salida es una página: la inclinación en las diez dimensiones, dos números de TCO, un puntaje de capacidad y una recomendación que el patrocinador ejecutivo puede firmar.

La disciplina es que el equipo se va con una recomendación, no con un menú. Decirle a un cliente que ambos son buenos y preguntarle qué prefiere devuelve la decisión justamente a quienes están menos preparados para tomarla. El costo por sí solo no debe decidir, porque el ítem de Red Hat es el componente menor y el costo de un acelerador mal elegido es mayor que la suscripción que ahorra. En el compromiso raro donde el taller termina genuinamente parejo, la respuesta es un piloto de treinta días de ambos, con dos equipos reales. De cualquier modo, el punto final es el mismo Horizon 3, así que una decisión revisada al final del primer año migra la internal developer platform y el clúster sin reconstruir la estrategia.

Referencias

Fuentes primarias para los dos aceleradores, la fundación compartida y el framework de decisión.

- Red Hat Developer Hub, la internal developer platform con soporte enterprise en el stack de Three Horizons.
- Azure Red Hat OpenShift, el sustrato de clúster operado en conjunto para Three Horizons.
- Backstage, el portal de desarrollador open-source que tu propio equipo opera en Open Horizons.
- Azure Kubernetes Service, el sustrato de clúster de Open Horizons.
- GitHub Copilot Documentation, agent mode, agentes personalizados y el model picker compartido en ambos aceleradores.
- Model Context Protocol Specification, el estándar abierto detrás de las 13 configuraciones de servidor MCP.

- Microsoft Entra, el plano único de identidad para personas y agentes en ambos aceleradores.
- The Forrester Wave: Platform Engineering Solutions, Q1 2026, referencia externa para la decisión de internal developer platform.

Paula Silva

AI-NATIVE SOFTWARE ENGINEER

in/paulanunes

v1.0.0 · 2026-07-02

Building the future of software development with AI and Agentic DevOps

04 LA EJECUCIÓN

La secuencia de 90/180/365 días: de la infraestructura bruta a los agentes en producción en un año.

La estrategia solo importa si llega a producción. Este capítulo convierte el argumento en un calendario: una secuencia de 90, 180 y 365 días que lleva a una empresa desde la infraestructura Azure bruta hasta agentes en producción dentro de un año. La secuencia respeta una regla por encima de todas. Construyes cada horizonte sobre el horizonte que está debajo, y te niegas a saltarte la fundación.

Días 0 a 90: construye la fundación H1

Los primeros noventa días producen una fundación que corre en producción, no una demo. Provisionas el sustrato con Terraform: un clúster AKS con Workload Identity, Azure Key Vault para secretos, ACR para imágenes, una VNet zero-trust y los network security groups que la acompañan. Sobre eso levantas Backstage con los primeros golden paths H1, para que un desarrollador pase de la intención a un servicio en cumplimiento en minutos. Argo CD conecta el GitOps con políticas de sincronización por entorno. Prometheus y Grafana encienden el dashboard de la plataforma. Tres equipos de aplicación se incorporan por los golden paths, y las Cuatro Métricas Clave DORA entran en vivo para los tres.

La meta para el día noventa es concreta: tiempo hasta el primer PR por debajo de un día, frente a una línea de base típica de dos a cuatro semanas. Dos cosas hacen que ese número sea real. La primera es el propio golden path, que codifica la mejor práctica actual de la empresa como una plantilla versionada, no como una página de wiki. La segunda es el archivo de instrucciones del repositorio. Un .github/copilot-instructions.md curado le dice a GitHub Copilot en agent mode cómo funciona realmente esta base de código, para que el agente construya dentro del golden path en lugar de adivinar. Aquí DORA no es un marcador. Es el filtro por el cual observas

el efecto de la IA en la entrega, así que instrumentas las métricas para medir, no para clasificar.

Meses 3 a 6: la mejora H2

Con la fundación creíble, los meses tres a seis la amplían. La cadena de suministro se endurece: imágenes firmadas, procedencia SLSA y controladores de admisión que rechazan cargas de trabajo no firmadas. Una malla de servicios (Istio en Open Horizons) se encarga de la gestión de tráfico, del mTLS y de la entrega progresiva. El catálogo de servicios supera el ochenta por ciento de cobertura de las cargas elegibles, y la documentación como código se vuelve el estándar en toda la organización para cada servicio nuevo. Aquí es donde los cuatro pilares de la CNCF terminan de conectarse: Golden Paths, Guardrails, Safety Nets y Manual Review.

El punto de H2 es que los mismos cuatro pilares ahora sirven a dos poblaciones. Los golden paths son la superficie de ejecución sancionada tanto para un desarrollador humano como para un agente. Los guardrails escritos para prevenir la mala configuración humana se vuelven el campo de contención de los permisos de runtime de un agente. Las safety nets que revierten un despliegue malo se vuelven los bucles de reconciliación que atrapan la inyección de prompts y los picos de costo. Para el día 180 la meta es tiempo hasta la primera producción de un nuevo microservicio por debajo de una semana, adopción de la plataforma por encima de la mitad de las cargas elegibles, y las métricas DORA moviéndose en la dirección correcta. La plataforma alcanza el nivel Scalable en la mayoría de las dimensiones de madurez de la CNCF.

Meses 6 a 12: la innovación H3

Los dos últimos trimestres ponen cargas de trabajo de IA en producción sobre el mismo sustrato. Entra la primera plantilla H3: una aplicación RAG o un Foundry agent. Los sistemas multi-agente se exploran mediante su propia plantilla. Cada plantilla H3 hereda los primitivos H1 y H2, así que un Foundry agent es estructuralmente un microservicio más un binding de modelo más un pipeline de evaluación, no una arquitectura separada. La observabilidad de agentes se

completa: trayectorias, costo, muestreo de salida y resultados de evaluación, todo en un único dashboard.

La prueba de todo el año está aquí. El agent mode consume los mismos golden paths que los desarrolladores. Un custom agent que construye un servicio corre la plantilla idéntica a la que correría un humano, satisface los guardrails idénticos y pasa por el gate de manual review idéntico. Corre contra un modelo del selector de Copilot, Claude Opus 4.8 para el razonamiento difícil y Sonnet 5 o Haiku 4.5 para el trabajo de rutina, y cada llamada se mide en GitHub AI Credits, donde un crédito vale un centavo de dólar. Los servidores MCP les dan a esos agentes acceso gobernado al catálogo y a las herramientas de la plataforma. Para el día 365, el tiempo hasta la primera producción de una carga de IA cayó de la línea de base de nueve a dieciocho meses a 90 a 180 días, y la flota de agentes creció de uno o dos agentes a diez o veinte en producción.

Los tres innegociables

Tres compromisos separan a las empresas que terminan esta secuencia de las que se estancan en el mes cuatro.

Trata la plataforma como un producto, no como un proyecto

Un proyecto tiene una fecha de término y un gerente de proyecto. Un producto no tiene ninguno de los dos; tiene un roadmap y un gerente de producto que lo posee. La distinción no es cosmética. El valor de la plataforma se compone a lo largo de años. La compresión de Forrester es un efecto de un año, y el múltiplo de ROI de McKinsey es un efecto de tres años, y ambos exigen que la plataforma exista como una cosa continua, con dueño y financiada.

Niégate a saltarte el H1

La presión por mostrar el H3 en noventa días es real, y está equivocada. Todo atajo que se salta la fundación reaparece después como deuda: regresiones de seguridad, plataformas sombra, putrefacción de contexto. La disciplina que lo evita es autoridad. El equipo de plataforma debe poder decir no al trabajo de H3 que depende de trabajo de H1 que aún no aterrizó, y el liderazgo debe defender ese no cuando el patrocinador presione.

Entrega el primer custom agent de Copilot temprano

Incluso un custom agent mínimo que responde cómo hago X en nuestra plataforma se paga dentro de un trimestre, porque absorbe la carga de onboarding que de otro modo caería sobre el equipo de plataforma. Entrégalo en la semana ocho a diez, no en la semana treinta. Es la primera prueba ejecutable de que la plataforma se está construyendo para agentes y humanos a la vez, y sus datos de trayectoria hacen mejor la siguiente versión.

> DEFINICIÓN
>
> La secuencia es el contrato: H1 en noventa días, H2 para el día 180, H3 para el día 365. Construye cada horizonte sobre el de abajo, trata la plataforma como un producto y entrega el primer custom agent temprano. Sáltate la fundación y no ahorras tiempo; aplazas la cuenta.

Referencias

Fuentes primarias para la secuencia por fases, la progresión de madurez y las métricas de entrega detrás de las metas de noventa días.

- DORA, 2025 Accelerate State of DevOps Report, las métricas de entrega: las Cuatro Métricas Clave DORA como filtro para el efecto de la IA, con los mejores ejecutores obteniendo entre 20 y 30 por ciento de ganancias de productividad.
- CNCF, Platform Engineering Maturity Model, la progresión de Operational a Scalable y a Optimizing que la secuencia sigue.
- Forrester, The Forrester Wave: Platform Engineering Solutions Q1 2026, la investigación de platform engineering: 77 por ciento de las empresas atribuyen mejoras medibles de time-to-market y 85 por ciento reportan impacto positivo en ingresos a plataformas internas.
- CNCF, Platform Engineering Whitepaper, los cuatro pilares y los golden paths que la secuencia conecta en H1 y H2.
- Backstage, open framework for building developer portals, el portal de desarrolladores y la capa de templating de golden paths usados desde el primer día.

- McKinsey & Company, The State of AI, la evidencia de ROI plurianual detrás de tratar la plataforma como un producto, no como un proyecto.

Paula Silva

AI-NATIVE SOFTWARE ENGINEER

in/paulanunes

v1.0.0 · 2026-07-02

Building the future of software development with AI and Agentic DevOps

05 EL RETORNO

Madurez, medición y ROI: qué es bueno de verdad, y cómo defenderlo.

Una plataforma que no se puede medir no se puede defender. Este capítulo entrega al equipo de plataforma tres capas de medición y un business case. El Modelo de Madurez de Platform Engineering de la CNCF ubica a la plataforma en una escalera de Provisional a AI-Native. Las Cuatro Métricas Clave DORA, extendidas con métricas de IA de 2026 y un conjunto separado de salud de agentes, exponen entrega y comportamiento de agentes en cuatro dashboards. La gobernanza de costo corre sobre GitHub AI Credits. Cierra con los dos números de campo y la evidencia externa que defienden la inversión a nivel del CFO.

El modelo de madurez de la CNCF: cinco dimensiones, cinco niveles

El Modelo de Madurez de Platform Engineering de la CNCF puntúa a la plataforma en cinco dimensiones independientes. Inversión pregunta cómo se financia y se dota de personal la plataforma. Adopción pregunta con qué amplitud y profundidad se usa. Interfaces pregunta cómo interactúan los usuarios con ella. Operaciones pregunta cómo se opera la propia plataforma. Medición pregunta qué mide el equipo de plataforma sobre sí mismo y sobre sus usuarios. Las dimensiones son independientes. Una plataforma puede estar bien financiada en Inversión y aun así encaminar a los desarrolladores por páginas de wiki en Interfaces. Tratar la madurez como cinco ejes separados es lo que mantiene honesta la puntuación.

Cada dimensión avanza por cinco niveles. Provisional es ad hoc y fragmentario. Operational significa que existe un esfuerzo dedicado, pero aún no productizado. Scalable significa que los usuarios se autoatienden por una interfaz estable con SLAs documentados. Optimizing significa que la dimensión produce métricas que impulsan su propia mejora. AI-Native significa que la dimensión es consumida por agentes a escala, sobre los mismos primitivos que sirven a los usuarios humanos. El

error común es promediar las cinco dimensiones en un único número halagüeño. La puntuación honesta es el mínimo. Una plataforma AI-Native en Adopción, pero Provisional en Inversión, es una plataforma Provisional.

AI-Native, nivel 5, cierra el Doble Mandato

AI-Native es el objetivo que cierra el Doble Mandato del capítulo 04. En el nivel 5, la Inversión lleva un presupuesto de doble mandato que financia tanto la IA que aumenta al equipo de plataforma como los primitivos de IA para los equipos de aplicación. La Adopción incluye agentes consumiendo los mismos Golden Paths que los desarrolladores humanos. Las Interfaces exponen esos Golden Paths mediante servidores MCP, de modo que la llamada "aprovisiona un servicio" de un agente invoca la misma plantilla que invocaría una persona. Las Operaciones son conducidas por agentes con revisión humana. La Medición abarca DORA, costo, ROI de IA y salud de agentes. Pocas empresas eran plenamente AI-Native a inicios de 2026. El objetivo realista es Optimizing en todas las dimensiones, con AI-Native al menos en Inversión y Adopción.

> DEFINICIÓN
>
> La puntuación honesta de madurez es el mínimo entre las cinco dimensiones, no el promedio. La dimensión más débil determina la madurez efectiva de la plataforma, porque la dimensión más débil es el modo de fallo que la IA amplifica primero.

Cuatro Métricas Clave DORA, las extensiones de IA y la salud de agentes

Las Cuatro Métricas Clave DORA siguen siendo la lente sobre el desempeño de entrega: frecuencia de despliegue, tiempo de entrega de cambios, tasa de fallos en cambios y tiempo de restauración del servicio. El informe DORA 2025, construido sobre casi 5.000 respuestas de encuesta, formula el hallazgo con claridad: la IA no arregla un equipo, amplifica lo que ya existe. Las Cuatro Métricas son el filtro por el cual esa amplificación se vuelve visible. En sistemas maduros la IA compone las métricas; en sistemas inmaduros la misma IA las regresa.

En 2026 el equipo de plataforma extiende DORA con cuatro familias específicas de IA. Uso de IA cubre el número de agentes en producción, el volumen de solicitudes y la tasa de error. Calidad de IA cubre la tasa de aprobación en evaluaciones y su deriva en el tiempo, más la fracción de salidas que un LLM-juez aprueba. Costo de IA cubre costo en créditos por carga de trabajo, latencia, tasa de acierto de caché y utilización por modelo. Comportamiento de IA cubre el cumplimiento de alcance de permisos y la completitud del rastro de auditoría. Estas familias son más útiles por carga de trabajo, no en el agregado. Una sola carga que regresa en el tiempo de entrega mientras el agregado mejora es el aviso temprano de que la plataforma está absorbiendo trabajo para el que aún no está lista.

La salud de agentes es un conjunto de métricas separado

La salud de agentes es distinta del desempeño de entrega. Un agente puede desplegar con frecuencia y fallar poco mientras su costo se desvía, su alcance se expande y sus salidas se vuelven inconsistentes. El conjunto mínimo de 2026 cubre cumplimiento de identidad (cada acción atribuible a una identidad gestionada, meta 100%), adherencia de alcance de permisos (ninguna acción fuera de alcance exitosa), costo por sesión (una distribución estable, con sesiones por encima de tres veces la mediana señaladas), detección de bucles, tasa de aprobación en evaluaciones, calidad de salida, tasa de error y detección de deriva. Todo esto aparece en los cuatro dashboards que ambos aceleradores entregan: plataforma, costo, uso de IA y seguridad. Captura continua de trayectoria en cada sesión, evaluación por muestreo y evaluación disparada ante anomalías mantienen la medición viable con miles de sesiones al día.

Gobernanza de costo en GitHub AI Credits

El uso de agentes de GitHub Copilot se mide en GitHub AI Credits. Desde el 1 de junio de 2026, un crédito equivale a un centavo de dólar estadounidense. Esa única conversión es lo que hace legible el gasto de agentes para el área financiera: cada prompt, llamada de herramienta y sesión de agente se resuelve en un conteo de créditos, y un conteo de créditos se resuelve en dólares. El dashboard de costo agrega esos créditos por agente, por carga de trabajo y por equipo. La atribución por agente es el prerrequisito para la familia de Costo de IA de la sección anterior.

Sin ella, el gasto de agentes se esconde dentro de una factura de nube agregada y el valor de negocio queda anecdótico.

La atribución habilita el control. Un circuit breaker de costo termina la sesión de un agente cuando su gasto en créditos cruza un umbral, el complemento en tiempo de ejecución del techo de presupuesto que los guardrails aplican antes de que empiece la sesión. El enrutamiento de modelo es la otra palanca. GitHub Copilot expone un selector de modelos (Claude Fable 5, Opus 4.8, Sonnet 5, Haiku 4.5, Gemini 3.1 Pro, GPT-5.x), así que el trabajo rutinario de agentes corre en Haiku 4.5 mientras el trabajo de alto juicio queda reservado para Opus 4.8 o Claude Fable 5. La utilización por modelo en el dashboard de uso de IA muestra si ese enrutamiento se sostiene, y la tasa de acierto de caché muestra si el contexto repetido se está pagando dos veces.

> DEFINICIÓN
>
> La atribución de GitHub AI Credits por agente, sumada a los circuit breakers de costo, convierte el gasto de agentes de una línea de nube sin control en un costo gobernado por sesión. Un crédito es un centavo de dólar; el dashboard de costo es donde esa aritmética se vuelve una conversación con el CFO.

El business case: qué es bueno de verdad, y cómo defenderlo

Dos números de campo anclan qué es bueno de verdad. El primero es adopción de golden path por encima del 60%, la porción de servicios nuevos que arrancan desde un Golden Path en lugar de desde cero. Por debajo de ese umbral el equipo de plataforma opera como un service desk; por encima, se vuelve un multiplicador de fuerza y los servicios paralelos desaparecen. Se sigue mensualmente y es responsabilidad del product manager de plataforma. El segundo es una ganancia mediana de experiencia del desarrollador (DevEx) de alrededor de 15 puntos en los dos trimestres tras la llegada de los Golden Paths, medida contra un baseline interno de antes y después. La mayor parte de esa ganancia está en carga cognitiva y feedback loops, no en velocidad bruta, y correlaciona con retención, no solo con productividad.

Esos dos números se conectan con resultados que un CFO reconoce. El State of AI 2025 de McKinsey, con más de 1.400 organizaciones, reporta que las organizaciones maduras en plataforma alcanzan valor de IA atribuible en cerca de 6 meses frente a cerca de 15 de los pares inmaduros, una compresión de 2,5x, a aproximadamente la mitad del costo por carga de trabajo. La investigación de platform engineering de Forrester constata 77% de las empresas atribuyendo mejoras medibles de time-to-market a plataformas internas y 85% reportando impacto positivo en ingresos. Los datos de IDC muestran la misma separación en el despliegue de agentes, con tasas de agentes en producción mucho más altas en organizaciones maduras en plataforma que en las inmaduras.

> DEFINICIÓN
>
> El business case pasa la prueba de defensibilidad cuando cada número está fundamentado o es directamente medible, el escenario pesimista aún produce valor neto positivo, y las métricas ya viven en los dashboards de la plataforma. Sé conservador en la proyección y entrega por encima de lo previsto en la ejecución.

Referencias

Fuentes primarias para el modelo de madurez, las capas de medición y el business case de este capítulo.

- CNCF Platform Engineering Maturity Model, el modelo de cinco dimensiones y cinco niveles contra el cual puntúa este capítulo.
- DORA 2025 Accelerate State of DevOps Report, las Cuatro Métricas y el hallazgo de la amplificación, sobre casi 5.000 respuestas de encuesta.
- McKinsey, The State of AI 2025, la compresión de ROI de 2,5x y aproximadamente la mitad del costo por carga de trabajo en organizaciones maduras en plataforma.
- Forrester Wave, Platform Engineering Solutions Q1 2026, 77% de mejoras de time-to-market y 85% de impacto positivo en ingresos a partir de plataformas internas.

- IDC, Agentic AI Platforms and Strategies, el diferencial de despliegue de agentes en producción entre organizaciones maduras e inmaduras en plataforma.
- GitHub Copilot Documentation, agent mode, el selector de modelos y el modelo de cobro en GitHub AI Credits que este capítulo mide.

Paula Silva

AI-NATIVE SOFTWARE ENGINEER

in/paulanunes

v1.0.0 · 2026-07-02

Building the future of software development with AI and Agentic DevOps