PLATFORM ENGINEERING PARA A EMPRESA AI-NATIVE · PLAYBOOK

A plataforma, não o modelo, decide se a IA corporativa entrega.

Três coisas são verdadeiras ao mesmo tempo em 2026: a adoção de IA é quase universal, a maioria dos projetos de IA ainda falha em produção, e os que têm sucesso compartilham uma plataforma madura por baixo. Este playbook defende que a fundação decide, mapeia a pirâmide de cinco camadas e o plano de controle de quatro pilares, entrega dois aceleradores sobre uma fundação Azure e GitHub, e transforma a estratégia numa sequência de 90/180/365 dias medida em DORA e GitHub AI Credits.

AUTORA

Paula Silva

CARGO

AI-Native Software Engineer

CONTATO

in/paulanunes

TEMPO DE LEITURA

~ 45 minutos, fim a fim

5

CAMADAS DA PIRÂMIDE

6

CAPÍTULOS

4

PILARES DA CNCF

2.5x

COMPRESSÃO DE ROI, PLATAFORMA MADURA

CAPÍTULOS

Chapter 00

Por que a fundação decide

Os três fatos de 2026, a tese da amplificação do DORA 2025, os cinco modos de falha quando a plataforma é fraca, e a evidência que um CTO leva ao conselho.

O Argumento →

Chapter 01

A Pirâmide de Cinco Camadas

Plataforma, Contexto, Cognição, Intenção e Agentic em ordem de dependência estrita, a regra de empilhamento que governa a construção, e onde o GitHub Copilot se encaixa em cada camada.

Arquitetura →

Chapter 02

Os Quatro Pilares como Plano de Controle

Golden Paths, Guardrails, Safety Nets e Manual Review reexpressos para governar agentes com a maquinaria que a plataforma já opera, vistos pela lente do Copilot e pelo Duplo Mandato de 2026.

Plano de Controle →

Chapter 03

Dois Aceleradores, Uma Fundação

Three Horizons no Red Hat Developer Hub e Open Horizons no Backstage, a fundação Azure e GitHub que não muda, e um método de um dia para escolher pelo arquétipo do cliente.

A Escolha →

Chapter 04

A sequência de 90/180/365 dias

O plano de execução em fases da infraestrutura Azure bruta aos agentes em produção em um ano, e os três inegociáveis que decidem se a sequência se concretiza.

Execução →

Chapter 05

Maturidade, medição e ROI

O modelo de maturidade da CNCF em cinco dimensões, as Quatro Métricas-Chave DORA mais as extensões de IA de 2026 e a saúde de agentes, governança de custo em GitHub AI Credits, e um business case defensável no nível do CFO.

O Retorno →

DICAS DE LEITURA

> **ATALHOS DE TECLADO**
>
> Aperte `/` para buscar. Use `j` e `k` para mover entre capítulos. Aperte `t` para alternar tema. Aperte `g` para ir ao topo.

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 O ARGUMENTO

Por que a fundação decide: a plataforma, não o modelo, define o teto.

Três coisas são verdadeiras ao mesmo tempo em 2026. A adoção de IA é quase universal. A maioria dos projetos de IA ainda falha em produção. E as organizações que têm sucesso compartilham um traço: uma plataforma madura por baixo. Este capítulo defende que a fundação, não o modelo, decide o resultado. Ele lê a evidência de 2025 e 2026 do jeito que um CTO precisa ler: como uma decisão de gasto com um risco mensurável.

Os três fatos de 2026

Comece pela adoção. O relatório The State of AI 2025 da McKinsey aponta que 78% das organizações já usam IA em pelo menos uma função. Nas empresas com que trabalho pela América Latina, essa adoção costuma ter um nome: GitHub Copilot, e cada vez mais o Copilot em agent mode rodando por todo o parque de repositórios. A Gartner projeta que 40% dos aplicativos corporativos vão embutir agentes para tarefas específicas até o fim de 2026, ante menos de 5% em 2024. A adoção está resolvida. A pergunta em aberto não é se as equipes usam IA. É o que essa IA faz com o sistema de entrega em que ela cai.

O segundo fato é menos confortável. O estudo The GenAI Divide, do MIT NANDA, publicado em 2025 com mais de 350 empresas, constatou que 95% dos pilotos de IA generativa não entregam valor mensurável. Falha ali é definida de forma operacional: o piloto nunca chegou a produção sustentada com resultados de negócio atribuíveis. O terceiro fato reconcilia os dois primeiros. As organizações que de fato chegam a produção são, com consistência marcante, as maduras em plataforma. O NANDA atribui a diferença não a modelos fracos, mas à ausência de contexto em nível de sistema e de laços de feedback. Isso é uma propriedade da

plataforma, não do modelo. A mesma ferramenta tem sucesso em um parque e falha no seguinte, e o que mudou foi a fundação por baixo.

A tese da amplificação

O mecanismo por trás desses três fatos tem nome, e a evidência mais forte é o DORA. O relatório DORA 2025 se apoia em quase 5 mil respostas de pesquisa e mais de 100 horas de dados qualitativos. Sua conclusão central é que a IA não se comporta como um insumo universal de produtividade. Ela se comporta como amplificadora do sistema de entrega que herda. Equipes em arquiteturas fracamente acopladas com feedback rápido convertem as mesmas ferramentas de IA em throughput real. Equipes em sistemas fortemente acoplados com feedback lento veem o volume extra de mudanças virar instabilidade. Mesmo insumo, resultados opostos. O DORA também relata que 90% das organizações pesquisadas adotaram ao menos uma plataforma interna, o que muda a conversa de se construir uma plataforma para quão madura é a sua.

É aqui que a lente do GitHub Copilot importa. Quando uma empresa liga o agent mode em milhares de repositórios, ela amplifica o sistema de entrega que esses repositórios já codificam. O trabalho da plataforma é alimentar o agente com contexto autoritativo para que a amplificação siga na direção certa. Na prática, isso significa um .github/copilot-instructions.md mantido em cada repositório, servidores MCP que expõem o catálogo de serviços, as políticas e os runbooks, e custom agents com escopo restrito a tarefas sancionadas. Também torna o custo visível. O uso do Copilot é cobrado como GitHub AI Credits, em que um crédito equivale a um centavo de dólar, e desde junho de 2026 essa linha mostra exatamente quanta reconstrução de contexto o parque está pagando. Uma fundação fraca faz esse número crescer. Uma madura o transforma em throughput.

> DEFINIÇÃO
>
> O DORA 2025 resume em uma linha: a IA não conserta um time; amplifica o que já está lá. Leia isso como uma regra de gasto. Investir em IA sem investir em plataforma é investir em amplificar o que já existe, na direção para a qual já aponta.

Cinco modos de falha quando a fundação é fraca

Quando a plataforma é subconstruída em relação à IA escalada sobre ela, cinco modos de falha se repetem. O primeiro é a dívida tripla. A IA gera dívida técnica por meio de código sem revisão, dívida cognitiva por meio de conhecimento que só existe em prompts e threads de chat, e dívida de intenção por meio de objetivos implícitos ou contraditórios. As três acumulam em paralelo e nenhuma resolve as outras. O segundo são as plataformas sombra. Quando a plataforma central é difícil de usar, cada equipe constrói a sua: pipelines sob medida, módulos de infraestrutura conflitantes, seus próprios servidores MCP e custom agents. Os dados de shadow IT da Gartner colocam esse gasto distribuído em duas a três vezes o custo de uma plataforma real, invisível em qualquer linha de orçamento.

O terceiro é o apodrecimento de contexto. Catálogos de serviços, documentação e linhagem ficam obsoletos em silêncio, e um agente que raciocina sobre um copilot-instructions.md desatualizado ou uma fonte MCP obsoleta devolve respostas plausíveis e erradas. No volume dos agentes, uma pequena taxa de apodrecimento vira uma grande taxa de erro. O quarto é a regressão de segurança. O Relatório Global de Ameaças 2026 da CrowdStrike documenta que organizações que implantaram assistentes de codificação com IA antes de amadurecer sua postura de segurança de plataforma viram um aumento de 38% em vulnerabilidades exploráveis nos primeiros doze meses. A regressão desaparece onde a plataforma aplica controles de cadeia de suprimentos e de identidade na admissão. O quinto é o problema do 100:1. Até 2028, a proporção agente-humano em uma empresa típica deve chegar a cerca de 100 para 1. A supervisão manual de identidade, credenciais e auditoria dos agentes não escala nessa proporção. Só uma plataforma que trata agentes como cargas de trabalho de primeira classe escala.

> DEFINIÇÃO
>
> Cinco modos de falha, uma causa: a IA roda mais rápido do que o processo ad hoc consegue governar. Dívida tripla. Plataformas sombra a duas ou três vezes o custo. Apodrecimento de contexto. Regressão de segurança. A proporção de 100:1 até 2028. Cada um é evitável pela plataforma. Nenhum é evitável por um modelo melhor.

A evidência que um CTO leva ao conselho

Para um executivo, os números decisivos são os que separam intenção de capacidade. A pesquisa da IDC sobre IA agêntica constata que, embora cerca de metade das organizações pesquisadas relate dez ou mais agentes em alguma forma de produção, menos de 7% rodam sequer um caso de uso em produção plena. A vontade está amplamente distribuída; a capacidade está concentrada nas organizações que construíram a plataforma primeiro. Essa única lacuna é mais útil do que qualquer manchete de produtividade, porque é observável dentro do seu próprio parque hoje. Conte quantos casos de uso de agentes estão de fato em produção plena, não em piloto. A resposta diz onde a sua fundação está.

O lado positivo também está documentado. A pesquisa de Platform Engineering da Forrester liga plataformas internas produtizadas diretamente a resultados de negócio: 77% das empresas atribuem ganhos mensuráveis de time-to-market à sua plataforma interna de desenvolvimento, e 85% relatam impacto positivo em receita. Junte os dois achados e o argumento se fecha. A restrição que limita a IA corporativa não é qual modelo está no seletor do Copilot. Claude Fable 5, Opus 4.8, Sonnet 5, Haiku 4.5, Gemini 3.1 Pro e a linha GPT-5.x estão todos a um clique no menu, e qualquer um deles vai amplificar o que herda. A restrição é a plataforma que os alimenta com contexto, governa suas ações e paga pelos seus tokens. É por isso que a fundação decide. Construa-a primeiro.

Referências

Fontes primárias para os três fatos, a tese da amplificação, os modos de falha e a evidência executiva.

- McKinsey, The State of AI 2025, a linha de base da adoção: 78% das organizações já usam IA em pelo menos uma função.
- MIT NANDA, The GenAI Divide: State of AI in Business 2025, a taxa de falha de 95% dos pilotos e seu diagnóstico como um problema de contexto e feedback.
- DORA 2025, Accelerate State of DevOps Report, a constatação da amplificação, apoiada em quase 5 mil respostas e mais de 100 horas de pesquisa qualitativa.
- Gartner, Over 40% of Agentic AI Projects Will Be Canceled by End of 2027, a projeção de cancelamentos movida por custo, valor indefinido e controles de

risco fracos.

- IDC, Agentic AI Platforms and Strategies, a lacuna de prontidão entre agentes declarados em produção e casos de uso de fato em produção plena.
- Forrester Wave, Platform Engineering Solutions Q1 2026, plataformas internas de desenvolvimento ligadas a time-to-market e impacto em receita.
- CNCF Platforms Whitepaper, a definição de referência das capacidades de plataforma que este capítulo defende.
- CrowdStrike, 2026 Global Threat Report, a regressão de segurança medida quando assistentes de codificação com IA chegam antes dos 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 A ARQUITETURA

A Pirâmide de Cinco Camadas: cinco camadas em ordem de dependência estrita, da plataforma aos agentes.

A empresa AI-native não é uma coisa só. É uma pirâmide de cinco camadas, cada uma dependendo da que está abaixo. O Platform Engineering é a base. Contexto, Cognição, Intenção e Agentic se empilham por cima, nessa ordem. A ordem não é conselho, é uma lei estrutural. Este capítulo mapeia as cinco camadas, nomeia o que cada uma detém, mostra onde o GitHub Copilot se encaixa e explica por que construir agentes sobre uma fundação ausente produz uma bagunça amplificada em vez de agentes.

A regra de empilhamento: cada camada fica em ordem de dependência estrita

A empresa AI-native é uma pirâmide, não um bolo de camadas. A camada da base é mais larga, custa mais e sustenta tudo o que está acima. Cada camada acima é mais estreita e explicitamente condicionada à camada de baixo. A ordem de construção não é preferência, é uma lei estrutural: você não pode construir a camada N+1 mais rápido do que a camada N. Você pode construí-las em paralelo, mas só se aceitar a dívida gerada pela lacuna. Uma pirâmide é montada de baixo para cima. Você não pode suspender o topo no ar e construir a base ao redor dele depois. A camada de Plataforma é construída primeiro porque toda camada acima dela precisa dos primitivos de plataforma para existir. Retire uma pedra e você não obtém uma pirâmide menor, você obtém um colapso.

A regra tem base empírica. O DORA 2025, construído sobre quase 5 mil respostas de pesquisa e mais de 100 horas de dados qualitativos, identificou que 90% das organizações adotaram ao menos uma plataforma interna, com correlação direta entre a qualidade da plataforma e o que a IA entrega. O relatório resume o mecanismo em uma linha: "AI doesn't fix a team; it amplifies what's already there".

Ou seja, a IA não conserta um time, ela amplifica o que já existe. Essa frase é a pirâmide reformulada. A IA consome a pilha de baixo para cima. Uma base fraca produz um topo fraco, por mais forte que o topo pareça isoladamente. As mesmas ferramentas de IA, resultados opostos, e a plataforma é a diferença.

L1 Plataforma a L5 Agentic: o que cada camada faz e o que cada uma detém

A Camada 1, Platform Engineering, é o substrato self-service, governado por políticas e observável. Ela detém os quatro pilares da CNCF: Golden Paths, Guardrails, Safety Nets e Manual Review. Golden Paths levam um desenvolvedor ou um agente de "quero construir X" a um sistema em conformidade e em execução, sem abrir um chamado. Guardrails são preventivos, aplicados na admissão em vez de detectados após o deploy. Safety Nets se recuperam das falhas que os guardrails não preveniram. Manual Review mantém um humano no loop para ações irreversíveis. Os artefatos concretos são um portal Backstage, GitOps, policy-as-code, observabilidade com OpenTelemetry e workload identity. A Camada 2, Contexto, é o conhecimento corporativo codificado em formato consumível por máquinas: o catálogo de serviços, documentação como código, linhagem, stores vetoriais e servidores MCP. Ela depende da L1 porque a plataforma detém os primitivos de identidade, observabilidade e ciclo de vida que tornam o contexto confiável.

A Camada 3, Cognição, são os modelos, embeddings e avaliação que consomem contexto para produzir saídas, mediados por um gateway de modelos que a plataforma provisiona e governa. A Camada 4, Intenção, são os objetivos, especificações e políticas codificados contra os quais os agentes planejam. Intenção sem contexto é alucinação; contexto sem intenção é descrição sem direção. A Camada 5, Agentic, são agentes autônomos e orientados a objetivos que combinam intenção, contexto e cognição para agir. Esta é a camada visível para a maior parte do investimento em IA em 2026, e é a camada com maior probabilidade de falhar quando as camadas abaixo estão ausentes. Um agente é um modelo mais um harness. O modelo é a parte plugável. O harness é tudo ao redor dele: ferramentas, memória, contexto, validação e gates. O harness é as quatro camadas inferiores em forma operacional, e é por isso que um agente em produção é o teste integrado de se a fundação está sólida.

A lente do Copilot: onde o GitHub Copilot se mapeia na pirâmide

O GitHub Copilot torna a pirâmide concreta. A camada de contexto é onde vivem os servidores MCP e o arquivo .github/copilot-instructions.md. Servidores MCP expõem sistemas corporativos (Azure, GitHub, monitoramento, tickets) ao Copilot por um único protocolo padrão, de modo que o agente lê a realidade da empresa em vez de adivinhá-la. O arquivo copilot-instructions é contexto no escopo do repositório: as convenções, a arquitetura e as regras que a base de código já codifica, escritas onde toda sessão do Copilot as lê. Ambos são artefatos da Camada 2. Nenhum é um modelo. São o conhecimento sobre o qual o modelo raciocina, e são o que impede um modelo capaz de produzir uma saída plausível que não corresponde a como a base de código realmente funciona.

A camada cognitiva é o seletor de modelos do Copilot. Em 2026 o seletor oferece Claude Fable 5, Opus 4.8, Sonnet 5, Haiku 4.5, Gemini 3.1 Pro e a família GPT-5.x. Nenhum modelo é o melhor para toda tarefa, então o seletor é o gateway de modelos tornado visível: você direciona um modelo rápido e barato para uma edição pequena e um modelo de raciocínio mais forte para um refactor difícil. O uso é cobrado em GitHub AI Credits, onde um crédito equivale a um centavo de dólar, na medição que entrou em vigor em 1 de junho de 2026. O agent mode do Copilot e os agentes customizados são a superfície da Camada 5: combinam as instruções (intenção e contexto), o modelo escolhido (cognição) e os controles da plataforma para agir sobre o repositório. O seletor sem as instruções e sem o MCP é cognição sem ancoragem.

Pular para o L5 sem o L1 a L4 não produz agentes

A falha mais comum é financiar a camada Agentic, que é visível e fácil de financiar, sem financiar as camadas de Plataforma e Contexto, que são invisíveis e sem glamour. O resultado é um agente que funciona em uma demo e falha em produção. Ele não tem contexto ancorado, então alucina. Ele não tem permissões governadas, então age fora do seu escopo. Ele não tem trajetórias observáveis, então ninguém consegue depurar ou certificar o que ele fez. A Gartner estima que 40% dos projetos de IA agêntica serão cancelados até 2027 por custo, valor indefinido e controles de risco fracos. Estruturalmente, esses são deploys de agentes

sobre fundações inadequadas: uma pirâmide com uma ou mais camadas inferiores ausentes.

Uma organização que pula para o L5 sem o L1 a L4 no lugar não produz agentes. Produz uma bagunça amplificada. A correção nunca é consertar o agente. É retrofitar as camadas ausentes, em ordem de dependência, o que custa mais depois que o agente já foi demonstrado a executivos do que teria custado construir corretamente da primeira vez. A versão Copilot do mesmo erro é entregar a um time o modelo mais forte do seletor sem nenhum arquivo copilot-instructions e sem nenhum servidor MCP conectado. O modelo não é a restrição. O contexto e a plataforma abaixo dele são. Comece pelo L1. Comprometa-se com a fundação antes dos agentes.

> DEFINIÇÃO
>
> A regra de empilhamento em uma linha: você não pode construir a camada N+1 mais rápido do que a camada N, e só pode construí-las em paralelo se estiver disposto a carregar a dívida gerada pela lacuna. Pular para o L5 sem o L1 a L4 não produz agentes. Produz uma bagunça amplificada.

Referências

Fontes primárias para a pirâmide de cinco camadas, a regra de empilhamento e as fundações de plataforma e contexto sobre as quais ela se apoia.

- CNCF Platforms Whitepaper, a definição de referência da camada de plataforma como um substrato self-service, governado por políticas e observável.
- CNCF Platform Engineering Maturity Model, a fonte dos quatro pilares que a Camada 1 detém: Golden Paths, Guardrails, Safety Nets e Manual Review.
- DORA 2025, State of AI-assisted Software Development, a base empírica da amplificação: a IA amplifica o que já existe.
- Backstage, o portal de desenvolvedor open-source que materializa os Golden Paths e o catálogo de serviços na base da pirâmide.

- Model Context Protocol Specification, o padrão aberto que os servidores MCP usam para expor sistemas corporativos como contexto consumível por agentes.
- Vishnyakova et al., From Context Engineering to Multi-Agent Architecture, a hierarquia de dependência de contexto a intenção que as camadas superiores da pirâmide formalizam.

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 O PLANO DE CONTROLE

Os Quatro Pilares como Plano de Controle: como uma plataforma governa agentes com a maquinaria que já opera.

Golden Paths, Guardrails, Safety Nets e Manual Review foram concebidos para governar a entrega humana. Em 2026 os mesmos quatro pilares governam agentes. A reexpressão é operacional, não metafórica. Uma plataforma que codificou os quatro pilares para desenvolvedores já detém a maior parte do plano de controle de IA. Este capítulo percorre os quatro pilares como a superfície que admite, contém, recupera e aprova cada ação que um agente executa.

Golden Paths e Guardrails: a superfície sancionada e o campo de contenção

Um agente é um serviço. Essa única equivalência é o que permite a uma plataforma governar agentes com a maquinaria que já opera. Um Golden Path é um fluxo de trabalho self-service e opinado que leva um desenvolvedor, ou um agente, de "quero construir X" a um sistema em conformidade e funcionando em minutos. Para agentes, o Golden Path se torna o scaffold pelo qual o agente é instanciado. A regra é direta: se não é um Golden Path, o agente não o executa. O Golden Path é a superfície de execução sancionada, e tudo fora dela está fora da superfície.

Guardrails são a outra metade. São preventivos, não detectivos. Um controle detectivo sinaliza uma violação depois do deploy, o que é lento demais quando os agentes agem mais rápido do que qualquer loop de revisão humana. Um controle preventivo rejeita a violação na admissão, antes de ela aterrissar. A mesma política de OPA Gatekeeper ou Kyverno que rejeita um manifesto Kubernetes não conforme rejeita uma configuração de agente não conforme. O mesmo Workload Identity que emite credenciais de curta duração para um serviço as emite para um agente. A ação errada não recebe um aviso. Ela é tornada impossível.

O campo de contenção do agente

Sobre os guardrails compartilhados, a plataforma adiciona quatro que são específicos de agentes. Um guardrail de escopo de permissão declara os dados que um agente pode ler e as ações que pode tomar, aplicado em tempo de execução. Um guardrail de filtro de saída redige e classifica antes de a saída chegar ao sistema consumidor. Um guardrail de residência de dados roteia o tráfego apenas para endpoints de modelo em regiões permitidas. Um guardrail de circuit breaker de custo encerra uma sessão quando o gasto cruza um teto. Juntos, esses quatro formam o campo de contenção do agente, aplicado na admissão e de novo em tempo de execução.

> DEFINIÇÃO
>
> Golden Paths são a única maneira sancionada de um agente começar. Guardrails são o que ele precisa satisfazer para continuar rodando. Um admite, o outro contém. Nenhum é opcional.

Safety Nets e Manual Review: recupere o reversível, gate o irreversível

Guardrails previnem. Safety Nets recuperam do que a prevenção deixou passar. O Safety Net mais fundamental é a reconciliação GitOps: Argo CD ou Flux comparam continuamente o estado implantado com o estado declarado no Git e corrigem o desvio sem humano no loop. Reexpressado para agentes, o mesmo loop se torna rollback a nível de decisão. Quando a saída de um agente se desvia de sua especificação declarada, quando o custo por tarefa sobe, ou quando a taxa de sucesso cai, o Safety Net reverte a decisão ou escala. A detecção de loop aborta uma sessão que se repete sem progresso. Um congelamento orientado por SLO paralisa uma população de agentes no instante em que um objetivo de serviço é violado, do mesmo modo que um canary paralisa um rollout ruim.

Manual Review é o último pilar e o que ganha importância à medida que todo o resto acelera. Sua função é inserir julgamento humano nas fronteiras que o justificam, e em nenhuma outra. Mudanças de rotina passam por Guardrails e Safety Nets sem toque. Ações irreversíveis batem em um gate de aprovação: uma migração de dados

em produção, uma expansão de escopo de permissão, um gasto acima de um limite. Para agentes, o gate crítico é a expansão de capacidade. Antes de um agente ganhar uma nova fonte de dados ou uma nova ação, um humano aprova. Esse é o equivalente em IA de um deploy elevado em produção, e é o único gate que uma plataforma madura nunca automatiza para fora.

O agente auditor

A fricção é intencional, mas a coleta de evidências não precisa ser manual. Um agente auditor lê a mudança proposta, recupera a entrada do catálogo de serviços, o histórico de deploy e de incidentes, o impacto de custo e o status de conformidade de políticas, e então monta um pacote de revisão estruturado. O humano gasta tempo na decisão, não na coleta dos fatos. A carga cognitiva sobre o revisor cai. A responsabilização permanece exatamente onde estava. Este é um exemplo limpo de IA aumentando os quatro pilares em vez de substituí-los.

A lente do Copilot: guardrails que um agente satisfaz em tempo de execução

Os quatro pilares permanecem abstratos até tocarem uma ferramenta que o desenvolvedor já usa. O GitHub Copilot em agent mode é onde eu os aterrisso. Um arquivo .github/copilot-instructions.md de um repositório é um guardrail escrito como texto. Ele declara as convenções, as fronteiras e os padrões sancionados que o agente precisa honrar em cada tarefa naquele repositório. Custom agents estreitam ainda mais a superfície: cada um é escopado a um domínio, carrega suas próprias instruções e só alcança as ferramentas e o contexto que você concede. Esse escopo é um guardrail de escopo de permissão expresso na camada do Copilot, e não na camada do cluster.

O MCP é como o agente alcança contexto e ferramentas sob governança. Um servidor MCP expõe um conjunto definido de recursos e ações, e a plataforma decide quais agentes se conectam a quais servidores. O modelo em si vem do seletor do Copilot, e o seletor é seu próprio tipo de guardrail: um agente só pode rodar em um modelo que a organização habilitou, o que hoje significa Claude Fable 5, Claude Opus 4.8, Claude Sonnet 5, Claude Haiku 4.5, Gemini 3.1 Pro ou um modelo GPT-5.x. Desde 1 de junho de 2026, o uso premium é cobrado em GitHub AI Credits,

onde um crédito equivale a um centavo de dólar. Essa superfície de cobrança é o guardrail de circuit breaker de custo na prática: o gasto é medido por requisição, atribuível por time e limitado antes de fugir do controle.

> DEFINIÇÃO
>
> Na lente do Copilot, o .github/copilot-instructions.md é o guardrail, o custom agent é o escopo de permissão, o MCP é a fronteira de contexto governada, e os GitHub AI Credits são o circuit breaker de custo. Os quatro pilares não são um diagrama. São configuração.

O Duplo Mandato de 2026

Os quatro pilares descrevem como uma plataforma governa agentes. O Duplo Mandato descreve o que um time de plataforma deve à empresa enquanto faz isso. O Mandato A é aumentar a própria plataforma com IA para velocidade interna. Agentes fazem triagem de alertas, rascunham correções de runbook, propõem mudanças de infraestrutura, revisam policy-as-code e geram invocações de Golden Path na IDE. O objetivo é que as melhorias de plataforma sejam entregues mais rápido, e cada melhoria que aterrissa mais cedo se compõe em todos os times que consomem a plataforma. A disciplina que mantém o Mandato A honesto é a adoção medida. Se menos de um terço das ações diárias do time realmente passa pelos agentes, as ferramentas são teatro, não integração.

O Mandato B é expor a IA como primitivos de plataforma que os times de aplicação consomem por meio de Golden Paths para velocidade externa. Inferência por um gateway de modelos governado, armazenamento vetorial, runtime de agentes, avaliação e observabilidade de agentes se tornam superfícies de primeira classe, do mesmo modo que Kubernetes e Postgres se tornaram de primeira classe uma década antes. O Azure AI Foundry é o gateway de referência nos dois aceleradores. O objetivo é que os times de aplicação entreguem produtos que usam IA em semanas, e não em trimestres. Uma plataforma que executa só o Mandato A produz um time de plataforma mais rápido que não move a empresa. Uma plataforma que executa só o Mandato B se torna um buffet de IA sem velocidade própria, e no fim o gargalo. Plataformas maduras executam os dois.

DEFINIÇÃO

O teste de maturidade é uma pergunta em duas metades. Você consegue nomear uma tarefa recorrente do time de plataforma que um agente absorveu, e um time de aplicação cujo cada primitivo de IA é um Golden Path da sua plataforma? Dois sins é uma plataforma madura. Um sim é metade de uma plataforma. Nenhuma das respostas é opcional em 2026.

Referências

Fontes primárias para os quatro pilares, a reexpressão do plano de controle de agentes e a superfície de execução do Copilot.

- CNCF Platform Engineering Maturity Model, a taxonomia de referência para Golden Paths, Guardrails, Safety Nets e Manual Review, e os níveis de maturidade por pilar.
- Backstage Software Templates (Scaffolder), o único ponto de entrada voltado ao desenvolvedor que torna um Golden Path a superfície de execução sancionada.
- Open Policy Agent Gatekeeper, aplicação de política no momento da admissão, o mecanismo que transforma um guardrail em um erro de compilação para a infraestrutura.
- Argo CD Documentation, reconciliação GitOps, o Safety Net fundamental que corrige o desvio de volta ao estado declarado.
- GitHub Copilot Documentation, a superfície de execução de referência para agent mode, custom agents e .github/copilot-instructions.md.
- Microsoft Entra Agent ID, identidade gerenciada de agente, o primitivo que permite aos mesmos guardrails governar agentes e serviços de forma uniforme.
- Azure AI Foundry Documentation, o gateway de modelos governado no centro do Mandato B.
- DORA Report 2025, a constatação da amplificação: a IA não conserta um time, ela amplifica o que a plataforma já é.

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 A ESCOLHA

Dois Aceleradores, Uma Fundação.

Os dois aceleradores chegam ao mesmo lugar. O Three Horizons e o Open Horizons constroem a mesma plataforma AI-Native, sobre a mesma fundação Azure e GitHub, e ambos terminam no mesmo Horizon 3. O que difere é o stack no meio: se a internal developer platform roda no Red Hat Developer Hub com suporte comercial, ou no Backstage que o seu próprio time opera. Este capítulo apresenta os dois, nomeia a fundação que não muda e entrega um método de um dia para escolher.

Three Horizons: Red Hat Developer Hub no Azure Red Hat OpenShift

O Three Horizons é o acelerador para clientes com investimento prévio em Red Hat ou exigência de suporte comercial. A internal developer platform é o Red Hat Developer Hub. O cluster é o Azure Red Hat OpenShift, com o control plane operado em conjunto pela Microsoft e pela Red Hat sob um SLA conjunto. O CI roda no OpenShift Pipelines, o GitOps no OpenShift GitOps, e a Red Hat mantém suporte enterprise em tempo integral. Setores regulados que exigem que cada componente da plataforma carregue sua própria certificação, FedRAMP, FIPS ou Common Criteria, costumam parar aqui, porque os componentes do stack da Red Hat detêm essas certificações e os equivalentes open-source não.

O pacote de assinaturas da Red Hat normalmente representa de 10 a 30 por cento do gasto total de plataforma, variando com o tamanho do cluster, o número de usuários e a cobertura de add-ons. Esse item de custo compra um modelo de suporte. A exigência interna do time de plataforma permanece operacional, rodar e personalizar a plataforma, em vez de engenharia profunda, porque as questões do stack upstream vão para a Red Hat, e não para os seus próprios desenvolvedores. Para um banco sob as regras do BACEN ou uma operadora de energia com uma frota RHEL em escala, o Red Hat Developer Hub sobre OpenShift é continuidade, não custo extra.

Open Horizons: Backstage no Azure através dos três horizontes

O Open Horizons é um único repositório template no GitHub que provisiona a mesma plataforma no Azure usando componentes open-source da CNCF. O conteúdo é concreto: 16 módulos Terraform para a fundação Azure Kubernetes Service, 22 templates de Golden Path distribuídos pelos três horizontes, 13 configurações de servidor MCP e 17 agentes custom do GitHub Copilot conectados para o Duplo Mandato. O Backstage é a internal developer platform, e o seu próprio time a opera. Um time que adota o Open Horizons não escreve uma plataforma do zero. Ele personaliza um ponto de partida opinado, o que representa uma diferença de ordem de magnitude em tempo, custo e risco.

A lente do GitHub Copilot é explícita aqui. Os 17 agentes custom rodam em agent mode; as convenções que cada agente segue vivem no .github/copilot-instructions.md; e as 13 configurações de servidor MCP dão a cada agente acesso tipado e limitado por papel ao Azure, ao GitHub, ao Terraform, ao Kubernetes e ao stack de observabilidade. O consumo dos agentes é cobrado em GitHub AI Credits, onde um crédito equivale a um centavo de dólar. A troca em relação ao Three Horizons é direta. Não há item de assinatura da Red Hat, e a exigência de capacidade interna é mais ampla, porque as questões upstream vão para a comunidade e para os seus próprios engenheiros, não para um SLA de fornecedor.

A fundação que não muda

Sob os dois aceleradores o substrato é idêntico, e vale nomeá-lo com precisão justamente porque é o que não muda. O Azure é a nuvem. O GitHub é a fonte de registro e a superfície de CI. Um único plano de identidade passa pelo Microsoft Entra, de modo que pessoas e agentes são governados no mesmo lugar. Um único model picker do GitHub Copilot expõe os mesmos modelos a cada desenvolvedor e a cada agente, independentemente do acelerador acima, incluindo Claude Opus 4.8 e Sonnet 5, Gemini 3.1 Pro, a linha GPT-5.x e opções mais leves como o Haiku 4.5 para trabalho barato e de alto volume. Os quatro pilares da CNCF, o modelo de maturidade H1, H2 e H3, e o Duplo Mandato são os mesmos nos dois.

É por isso que a decisão é uma escolha de stack, não uma escolha de estratégia. A estratégia não varia: a mesma pirâmide de cinco camadas, os mesmos quatro pilares, o mesmo Duplo Mandato, o mesmo caminho de 90 a 180 dias até o Horizon 3. O que varia é o meio do stack, a internal developer platform, o cluster, o substrato de CI, as ferramentas de cadeia de suprimentos e se as assinaturas da Red Hat estão no contrato. Os dois caminhos terminam no mesmo destino Azure AI Foundry, expondo os mesmos Golden Paths, os mesmos servidores MCP e os mesmos agentes.

> DEFINIÇÃO
>
> Mesmo destino, mesma fundação, stack diferente por cima. Azure para nuvem, GitHub para código e CI, um único plano de identidade Microsoft Entra para pessoas e agentes, um único model picker do GitHub Copilot para os dois. O acelerador que você escolhe muda o meio do stack, nunca a estratégia e nunca o ponto final.

Escolhendo pelo arquétipo do cliente

A maioria dos clientes se inclina antes de analisar, e a inclinação costuma estar certa. Um varejista digital-native já no Azure Kubernetes Service, com cultura OSS-first e um número de fornecedores deliberadamente pequeno, se inclina para o Open Horizons. Um banco regulado pelo BACEN com um parque OpenShift on-premise e uma cláusula de suporte comercial no procurement se inclina para o Three Horizons. A inclinação vem do perfil de procurement, da capacidade de engenharia, da postura regulatória e da estratégia de fornecedores, não de uma lista de recursos. Cerca de um quarto a um terço das empresas fica em um grupo intermediário onde nenhuma inclinação é dominante, e é aí que um método importa.

A descoberta de um dia

Para o grupo intermediário eu conduzo um workshop de descoberta de um dia. Duas semanas antes, o cliente envia um inventário: o parque Red Hat atual, o parque Kubernetes atual, o investimento em GitHub, o perfil do time de engenharia e a postura de procurement e regulatória. O dia tem cinco sessões de trabalho:

alinhamento de estratégia, uma passagem pela comparação de stack em dez dimensões, um modelo de custo total de propriedade de três anos construído com os números do próprio cliente, uma autoavaliação honesta da capacidade do time de operar o Backstage e o Azure Kubernetes Service upstream, e uma recomendação. A saída é uma página: a inclinação nas dez dimensões, dois números de TCO, uma pontuação de capacidade e uma recomendação que o patrocinador executivo pode assinar.

A disciplina é que o time sai com uma recomendação, não com um cardápio. Dizer a um cliente que os dois são bons e perguntar o que ele prefere devolve a decisão justamente a quem está menos preparado para tomá-la. O custo sozinho não deve decidir, porque o item da Red Hat é o componente menor e o custo de um acelerador mal escolhido é maior do que a assinatura que ele economiza. No engajamento raro em que o workshop termina genuinamente empatado, a resposta é um piloto de trinta dias dos dois, com dois times reais. De qualquer forma, o ponto final é o mesmo Horizon 3, então uma decisão revista ao fim do primeiro ano migra a internal developer platform e o cluster sem reconstruir a estratégia.

Referências

Fontes primárias para os dois aceleradores, a fundação compartilhada e o framework de decisão.

- Red Hat Developer Hub, a internal developer platform com suporte enterprise no stack do Three Horizons.
- Azure Red Hat OpenShift, o substrato de cluster operado em conjunto para o Three Horizons.
- Backstage, o portal de desenvolvedor open-source que o seu próprio time opera no Open Horizons.
- Azure Kubernetes Service, o substrato de cluster do Open Horizons.
- GitHub Copilot Documentation, agent mode, agentes custom e o model picker compartilhado nos dois aceleradores.
- Model Context Protocol Specification, o padrão aberto por trás das 13 configurações de servidor MCP.

- Microsoft Entra, o plano único de identidade para pessoas e agentes nos dois aceleradores.
- The Forrester Wave: Platform Engineering Solutions, Q1 2026, referência externa para a decisão 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 A EXECUÇÃO

A sequência de 90/180/365 dias: da infraestrutura bruta aos agentes em produção em um ano.

A estratégia só importa se entrar em produção. Este capítulo transforma o argumento em um calendário: uma sequência de 90, 180 e 365 dias que leva uma empresa da infraestrutura Azure bruta até agentes em produção dentro de um ano. A sequência respeita uma regra acima de todas. Você constrói cada horizonte sobre o horizonte abaixo dele, e se recusa a pular a fundação.

Dias 0 a 90: construa a fundação H1

Os primeiros noventa dias produzem uma fundação que roda em produção, não uma demo. Você provisiona o substrato com Terraform: um cluster AKS com Workload Identity, Azure Key Vault para segredos, ACR para imagens, uma VNet zero-trust e os network security groups que a acompanham. Sobre isso você sobe o Backstage com os primeiros golden paths H1, para que um desenvolvedor vá da intenção a um serviço em conformidade em minutos. O Argo CD conecta o GitOps com políticas de sincronização por ambiente. Prometheus e Grafana acendem o dashboard da plataforma. Três times de aplicação embarcam pelos golden paths, e as Quatro Métricas-Chave DORA entram no ar para os três.

A meta para o dia noventa é concreta: tempo até o primeiro PR abaixo de um dia, contra uma linha de base típica de duas a quatro semanas. Duas coisas tornam esse número real. A primeira é o próprio golden path, que codifica a melhor prática atual da empresa como um template versionado, não como uma página de wiki. A segunda é o arquivo de instrução do repositório. Um .github/copilot-instructions.md curado diz ao GitHub Copilot em agent mode como esta base de código realmente funciona, para que o agente construa dentro do golden path em vez de adivinhar. Aqui o DORA não é um placar. É o filtro pelo qual você observa o efeito da IA na entrega, então você instrumenta as métricas para medir, não para ranquear.

Meses 3 a 6: o aprimoramento H2

Com a fundação credível, os meses três a seis a ampliam. A cadeia de suprimentos é endurecida: imagens assinadas, procedência SLSA e controladores de admissão que rejeitam cargas de trabalho não assinadas. Uma malha de serviço (Istio no Open Horizons) cuida do gerenciamento de tráfego, do mTLS e da entrega progressiva. O catálogo de serviços passa de oitenta por cento de cobertura das cargas elegíveis, e a documentação como código vira o padrão em toda a organização para cada novo serviço. É aqui que os quatro pilares da CNCF terminam de ser conectados: Golden Paths, Guardrails, Safety Nets e Manual Review.

O ponto do H2 é que os mesmos quatro pilares agora servem a duas populações. Os golden paths são a superfície de execução sancionada tanto para um desenvolvedor humano quanto para um agente. Guardrails escritos para prevenir a má configuração humana viram o campo de contenção das permissões de runtime de um agente. Safety nets que revertem um deploy ruim viram os loops de reconciliação que pegam injeção de prompt e picos de custo. Até o dia 180 a meta é tempo até a primeira produção de um novo microsserviço abaixo de uma semana, adoção da plataforma acima de metade das cargas elegíveis, e as métricas DORA se movendo na direção certa. A plataforma alcança o nível Scalable na maioria das dimensões de maturidade da CNCF.

Meses 6 a 12: a inovação H3

Os dois últimos trimestres colocam cargas de trabalho de IA em produção sobre o mesmo substrato. O primeiro template H3 entra: uma aplicação RAG ou um Foundry agent. Sistemas multi-agente são explorados por meio do próprio template. Todo template H3 herda os primitivos H1 e H2, então um Foundry agent é estruturalmente um microsserviço mais um binding de modelo mais um pipeline de avaliação, não uma arquitetura separada. A observabilidade de agentes se completa: trajetórias, custo, amostragem de saída e resultados de avaliação, tudo em um único dashboard.

O teste do ano inteiro está aqui. O agent mode consome os mesmos golden paths que os desenvolvedores. Um custom agent que constrói um serviço roda o template idêntico ao que um humano rodaria, satisfaz os guardrails idênticos e passa pelo

gate de manual review idêntico. Ele roda contra um modelo do seletor do Copilot, Claude Opus 4.8 para o raciocínio difícil e Sonnet 5 ou Haiku 4.5 para o trabalho de rotina, e cada chamada é medida em GitHub AI Credits, onde um crédito vale um centavo de dólar. Servidores MCP dão a esses agentes acesso governado ao catálogo e às ferramentas da plataforma. Até o dia 365, o tempo até a primeira produção de uma carga de IA caiu da linha de base de nove a dezoito meses para 90 a 180 dias, e a frota de agentes cresceu de um ou dois agentes para dez ou vinte em produção.

Os três inegociáveis

Três compromissos separam as empresas que terminam esta sequência das que empacam no mês quatro.

Trate a plataforma como um produto, não como um projeto

Um projeto tem uma data de término e um gerente de projeto. Um produto não tem nenhum dos dois; tem um roadmap e um gerente de produto que o possui. A distinção não é cosmética. O valor da plataforma se compõe ao longo de anos. A compressão da Forrester é um efeito de um ano, e o múltiplo de ROI da McKinsey é um efeito de três anos, e ambos exigem que a plataforma exista como uma coisa contínua, com dono e financiada.

Recuse-se a pular o H1

A pressão para mostrar o H3 em noventa dias é real, e está errada. Todo atalho que pula a fundação reaparece depois como dívida: regressões de segurança, plataformas sombra, apodrecimento de contexto. A disciplina que evita isso é autoridade. O time de plataforma precisa poder dizer não ao trabalho de H3 que depende de trabalho de H1 que ainda não entrou, e a liderança precisa defender esse não quando o patrocinador pressionar.

Entregue o primeiro custom agent do Copilot cedo

Mesmo um custom agent mínimo que responde como eu faço X na nossa plataforma se paga dentro de um trimestre, porque absorve a carga de onboarding que de outra forma cairia sobre o time de plataforma. Entregue-o na semana oito a dez, não na

semana trinta. É a primeira prova executável de que a plataforma está sendo construída para agentes e humanos ao mesmo tempo, e os dados de trajetória dele tornam a próxima versão melhor.

> DEFINIÇÃO
>
> A sequência é o contrato: H1 em noventa dias, H2 até o dia 180, H3 até o dia 365. Construa cada horizonte sobre o de baixo, trate a plataforma como um produto e entregue o primeiro custom agent cedo. Pule a fundação e você não economiza tempo; você adia a conta.

Referências

Fontes primárias para a sequência em fases, a progressão de maturidade e as métricas de entrega por trás das metas de noventa dias.

- DORA, 2025 Accelerate State of DevOps Report, as métricas de entrega: as Quatro Métricas-Chave DORA como filtro para o efeito da IA, com performers de elite obtendo de 20 a 30 por cento de ganhos de produtividade.
- CNCF, Platform Engineering Maturity Model, a progressão de Operational para Scalable e para Optimizing que a sequência acompanha.
- Forrester, The Forrester Wave: Platform Engineering Solutions Q1 2026, a pesquisa de platform engineering: 77 por cento das empresas atribuem ganhos mensuráveis de time-to-market e 85 por cento relatam impacto positivo em receita a plataformas internas.
- CNCF, Platform Engineering Whitepaper, os quatro pilares e os golden paths que a sequência conecta em H1 e H2.
- Backstage, open framework for building developer portals, o portal de desenvolvedores e a camada de templating de golden paths usados desde o primeiro dia.
- McKinsey & Company, The State of AI, a evidência de ROI plurianual por trás de tratar a plataforma como um produto, não como um projeto.

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 O RETORNO

Maturidade, medição e ROI: o que é bom de verdade, e como defender isso.

Uma plataforma que não pode ser medida não pode ser defendida. Este capítulo entrega ao time de plataforma três camadas de medição e um business case. O Modelo de Maturidade de Platform Engineering da CNCF posiciona a plataforma numa escada de Provisional a AI-Native. As Quatro Métricas-Chave DORA, estendidas com métricas de IA de 2026 e um conjunto separado de saúde de agentes, expõem entrega e comportamento de agentes em quatro dashboards. A governança de custo roda em GitHub AI Credits. Fecha com os dois números de campo e a evidência externa que defendem o investimento no nível do CFO.

O modelo de maturidade da CNCF: cinco dimensões, cinco níveis

O Modelo de Maturidade de Platform Engineering da CNCF pontua a plataforma em cinco dimensões independentes. Investimento pergunta como a plataforma é financiada e composta. Adoção pergunta com que amplitude e profundidade ela é usada. Interfaces pergunta como os usuários interagem com ela. Operações pergunta como a própria plataforma é operada. Medição pergunta o que o time de plataforma mede sobre si mesmo e sobre seus usuários. As dimensões são independentes. Uma plataforma pode ser bem financiada em Investimento e ainda assim encaminhar desenvolvedores por páginas de wiki em Interfaces. Tratar a maturidade como cinco eixos separados é o que mantém a pontuação honesta.

Cada dimensão avança por cinco níveis. Provisional é ad hoc e fragmentário. Operational significa que existe um esforço dedicado, mas ainda não produtizado. Scalable significa que os usuários se autoatendem por uma interface estável com SLAs documentados. Optimizing significa que a dimensão produz métricas que dirigem sua própria melhoria. AI-Native significa que a dimensão é consumida por

agentes em escala, sobre os mesmos primitivos que servem usuários humanos. O erro comum é tirar a média das cinco dimensões num único número lisonjeiro. A pontuação honesta é o mínimo. Uma plataforma AI-Native em Adoção, mas Provisional em Investimento, é uma plataforma Provisional.

AI-Native, nível 5, fecha o Duplo Mandato

AI-Native é o alvo que fecha o Duplo Mandato do capítulo 04. No nível 5, o Investimento carrega um orçamento de duplo mandato que financia tanto a IA aumentando o time de plataforma quanto os primitivos de IA para times de aplicação. A Adoção inclui agentes consumindo os mesmos Golden Paths que os desenvolvedores humanos. As Interfaces expõem esses Golden Paths por servidores MCP, então a chamada "provisione um serviço" de um agente invoca o mesmo template que uma pessoa invocaria. As Operações são conduzidas por agentes com revisão humana. A Medição abrange DORA, custo, ROI de IA e saúde de agentes. Poucas empresas eram plenamente AI-Native no início de 2026. O alvo realista é Optimizing em todas as dimensões, com AI-Native ao menos em Investimento e Adoção.

> DEFINIÇÃO
>
> A pontuação honesta de maturidade é o mínimo entre as cinco dimensões, não a média. A dimensão mais fraca determina a maturidade efetiva da plataforma, porque a dimensão mais fraca é o modo de falha que a IA amplifica primeiro.

Quatro Métricas-Chave DORA, as extensões de IA e a saúde de agentes

As Quatro Métricas-Chave DORA continuam sendo a lente sobre o desempenho de entrega: frequência de deployment, lead time para mudanças, taxa de falha em mudanças e tempo para restaurar serviço. O relatório DORA 2025, construído sobre quase 5 mil respostas de pesquisa, formula o achado com clareza: a IA não conserta um time, amplifica o que já existe. As Quatro Métricas são o filtro pelo qual essa amplificação se torna visível. Em sistemas maduros a IA compõe as métricas; em sistemas imaturos a mesma IA as regride.

Em 2026 o time de plataforma estende o DORA com quatro famílias específicas de IA. Uso de IA cobre o número de agentes em produção, o volume de requisições e a taxa de erro. Qualidade de IA cobre a taxa de aprovação em avaliações e seu desvio ao longo do tempo, além da fração de saídas que um LLM-juiz aprova. Custo de IA cobre custo em créditos por carga de trabalho, latência, taxa de acerto de cache e utilização por modelo. Comportamento de IA cobre a conformidade de escopo de permissão e a completude da trilha de auditoria. Essas famílias são mais úteis por carga de trabalho, não no agregado. Uma única carga regredindo no lead time enquanto o agregado melhora é o aviso antecipado de que a plataforma está absorvendo trabalho para o qual ainda não está pronta.

A saúde de agentes é um conjunto de métricas separado

A saúde de agentes é distinta do desempenho de entrega. Um agente pode fazer deploy com frequência e falhar pouco enquanto seu custo desvia, seu escopo se expande e suas saídas ficam inconsistentes. O conjunto mínimo de 2026 cobre conformidade de identidade (toda ação atribuível a uma identidade gerenciada, meta 100%), aderência de escopo de permissão (nenhuma ação fora de escopo bem-sucedida), custo por sessão (uma distribuição estável, com sessões acima de três vezes a mediana sinalizadas), detecção de loops, taxa de aprovação em avaliações, qualidade de saída, taxa de erro e detecção de desvio. Tudo isso aparece nos quatro dashboards que ambos os aceleradores entregam: plataforma, custo, uso de IA e segurança. Captura contínua de trajetória em cada sessão, avaliação por amostragem e avaliação disparada em anomalias mantêm a medição viável em milhares de sessões por dia.

Governança de custo em GitHub AI Credits

O uso de agentes do GitHub Copilot é medido em GitHub AI Credits. Desde 1 de junho de 2026, um crédito equivale a um centavo de dólar dos EUA. Essa única conversão é o que torna o gasto de agentes legível para a área financeira: cada prompt, chamada de ferramenta e sessão de agente resolve para uma contagem de créditos, e uma contagem de créditos resolve para dólares. O dashboard de custo agrega esses créditos por agente, por carga de trabalho e por time. A atribuição por agente é o pré-requisito para a família de Custo de IA da seção anterior. Sem ela, o

gasto de agentes se esconde dentro de uma fatura de nuvem agregada e o valor de negócio fica anedótico.

A atribuição habilita o controle. Um circuit breaker de custo encerra a sessão de um agente quando seu gasto em créditos cruza um limite, o complemento em tempo de execução do teto de orçamento que os guardrails aplicam antes de a sessão começar. O roteamento de modelo é a outra alavanca. O GitHub Copilot expõe um seletor de modelos (Claude Fable 5, Opus 4.8, Sonnet 5, Haiku 4.5, Gemini 3.1 Pro, GPT-5.x), então o trabalho rotineiro de agentes roda em Haiku 4.5 enquanto o trabalho de alto julgamento fica reservado para Opus 4.8 ou Claude Fable 5. A utilização por modelo no dashboard de uso de IA mostra se esse roteamento se sustenta, e a taxa de acerto de cache mostra se o contexto repetido está sendo pago duas vezes.

> DEFINIÇÃO
>
> A atribuição de GitHub AI Credits por agente, somada aos circuit breakers de custo, transforma o gasto de agentes de uma linha de nuvem sem controle em um custo governado por sessão. Um crédito é um centavo de dólar; o dashboard de custo é onde essa aritmética vira uma conversa com o CFO.

O business case: o que é bom de verdade, e como defender

Dois números de campo ancoram o que é bom de verdade. O primeiro é adoção de golden path acima de 60%, a parcela de novos serviços que começam a partir de um Golden Path em vez de do zero. Abaixo desse limite o time de plataforma opera como um service desk; acima dele, vira um multiplicador de força e os serviços paralelos desaparecem. É acompanhado mensalmente e é responsabilidade do product manager de plataforma. O segundo é um ganho mediano de experiência do desenvolvedor (DevEx) de cerca de 15 pontos nos dois trimestres após os Golden Paths entrarem no ar, medido contra um baseline interno de antes e depois. A maior parte desse ganho está em carga cognitiva e feedback loops, não em velocidade bruta, e correlaciona com retenção, não só com produtividade.

Esses dois números conectam-se a resultados que um CFO reconhece. O State of AI 2025 da McKinsey, com mais de 1.400 organizações, relata que organizações maduras em plataforma alcançam valor de IA atribuível em cerca de 6 meses contra cerca de 15 dos pares imaturos, uma compressão de 2,5x, a aproximadamente metade do custo por carga de trabalho. A pesquisa de platform engineering da Forrester constata 77% das empresas atribuindo ganhos mensuráveis de time-to-market a plataformas internas e 85% relatando impacto positivo em receita. Os dados da IDC mostram a mesma separação na implantação de agentes, com taxas de agentes em produção bem mais altas em organizações maduras em plataforma do que nas imaturas.

> DEFINIÇÃO
>
> O business case passa no teste de defensibilidade quando todo número é fundamentado ou diretamente mensurável, o cenário pessimista ainda produz valor líquido positivo, e as métricas já vivem nos dashboards da plataforma. Seja conservador na projeção e entregue acima do previsto na execução.

Referências

Fontes primárias para o modelo de maturidade, as camadas de medição e o business case deste capítulo.

- CNCF Platform Engineering Maturity Model, o modelo de cinco dimensões e cinco níveis contra o qual este capítulo pontua.
- DORA 2025 Accelerate State of DevOps Report, as Quatro Métricas e o achado da amplificação, sobre quase 5 mil respostas de pesquisa.
- McKinsey, The State of AI 2025, a compressão de ROI de 2,5x e aproximadamente metade do custo por carga de trabalho em organizações maduras em plataforma.
- Forrester Wave, Platform Engineering Solutions Q1 2026, 77% de ganhos de time-to-market e 85% de impacto positivo em receita a partir de plataformas internas.

- IDC, Agentic AI Platforms and Strategies, o diferencial de implantação de agentes em produção entre organizações maduras e imaturas em plataforma.
- GitHub Copilot Documentation, agent mode, o seletor de modelos e o modelo de cobrança em GitHub AI Credits que este capítulo mede.

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