

Resenha do artigo Thoughtworks Technology Radar

Aluno: Douglas Galvão Souza Machado

Disciplina: Projeto de Software

Professor: João Paulo Aramuni

Thoughtworks Technology Radar com foco no estágio "Adote"

O artigo *Technology Radar*, 30º volume, publicado pela empresa Thoughtworks, se autointitula como “um guia de opinião sobre o universo de tecnologia atual”. Esse documento é dividido em duas partes: a primeira abrange *Temas* relevantes e a segunda o *Radar* propriamente dito. Na primeira parte, são discutidos tópicos abordados durante a última reunião do Conselho Consultivo de Tecnologia (em inglês, Technology Advisory Board - TAB), um grupo formado por 22 tecnologistas experientes da Thoughtworks. Já a segunda parte do artigo é composta por um radar que aborda Técnicas, Ferramentas, Plataformas e Linguagens/Frameworks, categorizando-as em quatro estágios: *Evite*, *Avalie*, *Experimente* e *Adote*. Para fins desta resenha, vamos focar apenas nos blips que estão no estágio *Adote*.

Um *blip* é uma tecnologia ou técnica que desempenha um papel fundamental no desenvolvimento de software. Esses blips estão constantemente em movimento no radar, ou seja, não permanecem fixos em uma mesma posição. O artigo destaca que a confiança em recomendá-los aumenta à medida que avançam de estágio, evoluindo de *Experimente* para *Adote*.

Sobre os temas:

O **ecossistema de código aberto**, que tradicionalmente se baseia em licenças padronizadas pela Open Source Initiative (OSI), enfrenta atualmente um período de incerteza. A mudança abrupta de algumas ferramentas populares de licenças open source para modelos comerciais levanta sérias preocupações sobre a sustentabilidade de projetos que se tornaram pilares da comunidade. Embora cobrar por funcionalidades adicionais seja aceitável, a prática de bloquear a funcionalidade central de uma ferramenta essencial por trás de um *paywall* é uma questão preocupante, especialmente em ferramentas de Inteligência Artificial (IA). A Thoughtworks recomenda cautela e uma análise detalhada das licenças, prestando especial atenção aos termos e condições, para garantir que os componentes cruciais permaneçam sob licenças adequadas.

Nesse cenário de incertezas, as **Inteligências Artificiais (IAs)** dominam as discussões sobre o futuro do desenvolvimento de software. Ferramentas como GitHub Copilot, CodiumAI e Continue têm o potencial de revolucionar a forma como o código é escrito, revisado e testado. A integração de IA em todas as etapas do desenvolvimento pode melhorar a produtividade, a qualidade e a segurança do software. No entanto, há riscos. A dependência excessiva dessas ferramentas pode levar à negligência de boas práticas de engenharia, além de apresentar vulnerabilidades, especialmente em termos de segurança. O artigo defende a adoção pragmática e responsável de IAs, equilibrando seus benefícios com uma visão crítica de seus riscos.

Os **Modelos de Linguagem de Grande Porte (LLMs)** também estão em destaque, impulsionando a criação de padrões arquiteturais específicos para lidar com seus desafios e oportunidades. Ferramentas como *NeMo Guardrails* (para governança de LLMs) e *Langfuse* (para aumentar a observabilidade) são mencionadas, juntamente com a técnica de **Geração Aumentada por Recuperação (RAG)** para melhorar a qualidade das respostas geradas. A Thoughtworks destaca a importância de definir padrões e antipadrões à medida que essas tecnologias amadurecem e se consolidam no desenvolvimento de software.

Outro tema relevante é o afastamento da prática de **Integração Contínua (CI)** devido ao crescente uso de *Pull Requests* (PRs). Embora os PRs sejam úteis para revisão de código, podem se tornar gargalos no processo de desenvolvimento. O artigo sugere soluções, como ferramentas de automação e métricas, para alinhar os PRs com os princípios da CI, garantindo que o processo de integração seja mais fluido.

Sobre o radar:

Técnicas:

- **Geração Aumentada por Recuperação (RAG):** Técnica usada para aprimorar as respostas geradas por LLMs, combinando a pergunta do usuário com informações recuperadas de fontes externas, como bancos de dados. A RAG enriquece o contexto do LLM, minimizando a chance de "alucinações" — respostas incorretas ou irrelevantes. No entanto, enfrenta desafios, como a janela de contexto limitada dos LLMs e a divisão eficiente de documentos extensos.

Plataformas:

- **CloudEvents:** Um padrão para eventos em arquiteturas modernas que visa melhorar a interoperabilidade entre diferentes provedores de nuvem. O CloudEvents facilita a integração entre sistemas, padronizando a troca de

eventos. Seu suporte pela Cloud Native Computing Foundation (CNCF) é uma prova de sua maturidade e da ampla adoção pela comunidade.

Ferramentas:

- **Conan:** Ferramenta de código aberto para o gerenciamento de dependências em projetos C/C++. O Conan simplifica o processo de gerenciamento de bibliotecas, garantindo consistência e reprodutibilidade em ambientes diversos. Ele é integrado com ferramentas populares, como CMake, e oferece suporte multiplataforma, o que o torna essencial para projetos complexos.
- **Kaniko:** Uma ferramenta voltada para a construção de imagens de container em ambientes Kubernetes, sem a necessidade de um *daemon* Docker. Isso aumenta a segurança e a eficiência, especialmente em pipelines de CI/CD. A sua flexibilidade e performance têm feito com que o Kaniko seja amplamente adotado como padrão em diversos cenários de desenvolvimento de containers.
- **Karpenter:** Operador Kubernetes para escalabilidade de nós. Ele se destaca por sua inteligência na análise de *workloads* e otimização de recursos. Ao contrário do Cluster Autoscaler tradicional, o Karpenter oferece uma abordagem mais dinâmica e eficiente para o provisionamento de recursos, ajustando a capacidade do cluster conforme a demanda. Sua adoção está crescendo, principalmente em plataformas como AWS EKS e Azure.

Conclusão:

O **Technology Radar** da Thoughtworks oferece uma visão clara e atualizada sobre as tendências tecnológicas e as ferramentas que estão moldando o futuro do desenvolvimento de software. A empresa demonstra uma abordagem crítica e cuidadosa ao recomendar tecnologias para adoção, enfatizando a necessidade de se equilibrar inovação com responsabilidade, especialmente em áreas como inteligência artificial e integração contínua. As recomendações no estágio *Adote* são bem fundamentadas e oferecem soluções práticas para desafios emergentes. Contudo, é importante destacar que a rápida evolução dessas tecnologias exige uma vigilância constante por parte das equipes de desenvolvimento, para que possam se adaptar e evoluir de acordo com as mudanças no cenário tecnológico.