

# Resenha Thoughtworks Technology Radar

Aluna: Ana Flávia de Carvalho Santos

## Introdução

O Radar da Thoughtworks é um guia valioso para entusiastas de tecnologia e desenvolvedores que buscam se manter atualizados em meio à constante evolução do cenário tecnológico. Mais do que apenas listar tecnologias emergentes, o Radar oferece uma análise criteriosa e categorizada, guiando a tomada de decisão sobre quais tendências adotar, experimentar ou evitar.

A estrutura do Radar é intuitiva e organizada em quatro anéis principais: Adote, Experimente, Avalie e Evite. Essa classificação indica o nível de confiança e maturidade que a Thoughtworks atribui a cada tecnologia listada. As ferramentas e práticas posicionadas no anel "Adote" são consideradas sólidas e prontas para serem amplamente utilizadas, enquanto aquelas em "Experimente" demonstram potencial, mas ainda exigem cautela e experimentação. "Avalie" sugere acompanhar de perto o desenvolvimento da tecnologia, e "Evite" alerta para riscos e desafios que exigem atenção.

A seleção das tecnologias presentes no Radar não se baseia em modismos passageiros, mas sim em uma análise profunda realizada pelo Conselho Consultivo de Tecnologia da Thoughtworks (TAB), que considera diversos fatores como maturidade, aplicabilidade em projetos reais e feedback de clientes. Além disso, o Radar se aprofunda em 4 temas relevantes que moldam o cenário tecnológico. Essas discussões abordam tendências e desafios específicos, como a crescente influência da Inteligência Artificial no desenvolvimento de software, as mudanças nas licenças de código aberto e a busca por otimizar práticas de Integração Contínua. Ao analisar esses temas, o Radar fornece um contexto estratégico para a tomada de decisão na produção de um software.

O Radar faz essa pesquisa para 4 contextos diferentes: Técnicas, Plataformas, Ferramentas e Linguagens e Frameworks. Nas próximas seções serão analisadas as tecnologias de cada uma dessas áreas que estão inseridas no anel "Adote", ou seja, que são fortemente sugeridas.

## Técnicas

### Geração Aumentada por Recuperação (RAG)

Na seção de Técnicas do Radar da Thoughtworks há a forte recomendação para adoção da Geração Aumentada por Recuperação (RAG). Essa técnica surge como uma

solução poderosa para um dos principais desafios na aplicação de Modelos de Linguagem Grandes, os LLMs: a geração de respostas precisas e contextualmente relevantes, que fujam das “alucinações”, ou seja, das respostas irrelevantes.

A RAG se destaca por sua capacidade de conectar o conhecimento implícito nos LLMs com informações específicas do contexto de cada aplicação. Através da integração com bancos de dados otimizados para pesquisa, a técnica permite que os LLMs acessem e processem informações relevantes de documentos, páginas web e outras fontes de dados, complementando o conhecimento adquirido durante seu treinamento. Ela reduz significativamente as chances de respostas imprecisas ou "alucinações", tornando os LLMs mais confiáveis para tarefas como pesquisa semântica, geração de conteúdo e chatbots.

## Plataformas

### CloudEvents

A ascensão das arquiteturas orientadas a eventos e microserviços trouxe consigo o desafio da interoperabilidade entre diferentes plataformas e provedores de nuvem. O CloudEvents, promovido pela Cloud Native Computing Foundation (CNCF), surge como uma solução robusta para este desafio, padronizando a forma como os eventos são descritos e compartilhados entre diferentes sistemas.

Com essa plataforma, desenvolvedores e arquitetos podem projetar e implementar sistemas distribuídos com maior confiança, sabendo que a comunicação entre os componentes será consistente e confiável, independentemente da linguagem de programação, plataforma ou infraestrutura utilizada. Essa padronização facilita a integração entre diferentes serviços e acelera o desenvolvimento de soluções complexas, tornando o CloudEvents uma escolha estratégica para empresas que buscam construir arquiteturas flexíveis.

## Ferramentas

### Conan

Classificado como uma ferramenta open source e voltado para projetos C/C++, o Conan simplifica o gerenciamento de dependências, muitas vezes uma tarefa complexa e com grande chance de erros. Com ele, as equipes podem definir, buscar e gerenciar facilmente as bibliotecas necessárias, garantindo a consistência entre diferentes plataformas e reduzindo o tempo de compilação de forma significativa. Essa otimização contribui para um desenvolvimento mais ágil e confiável, especialmente

em projetos de larga escala que podem ser muito complexos para a questão das dependências.

### Kaniko

Essa ferramenta se tornou essencial para equipes que utilizam Kubernetes e buscam uma alternativa robusta ao Docker para a construção de imagens de contêineres, já que o Kubernetes deixou de oferecer suporte a ele. Em um ambiente onde segurança e eficiência são cruciais, o Kaniko se destaca por sua capacidade de construir imagens de forma segura e eficiente em pipelines automatizados, sem a necessidade da instalação do Docker. Essa abordagem simplifica o processo de construção e implantação, tornando-o mais seguro e escalável.

### Karpenter

O Kubernetes possui uma funcionalidade chamada autoscaling horizontal, que permite ajustar automaticamente a quantidade de recursos alocados para uma aplicação conforme a demanda, mas essa capacidade depende da existência de nós. Enquanto o Cluster Autoscaler, solução nativa do Kubernetes, oferece uma capacidade básica de escalonamento de nós, o Karpenter surge como uma alternativa mais inteligente, monitorando continuamente os workloads em execução, analisando as restrições de cada aplicação e provisionando os tipos de nós mais adequados para atender às demandas, sem a necessidade de gerenciar a infraestrutura do cluster manualmente. O Karpenter automaticamente remove os nós excedentes, otimizando o uso de recursos.

## Linguagens e Frameworks

Para as linguagens e frameworks não há nada no radar no anel de “Adote”, portanto as ferramentas citadas são do anel “Experimente”.

### Astro

Está ganhando notoriedade como um framework de desenvolvimento web com foco em desempenho, útil para sites com grande quantidade de conteúdo estático, como blogs e páginas de marketing. Sua abordagem, que combina SSR e uma estrutura de "ilhas" que permite a inclusão de componentes JavaScript interativos apenas quando necessário, resulta em páginas extremamente rápidas e otimizadas. A facilidade de aprendizado, devido à sua sintaxe baseada em HTML, torna o Astro ainda mais atraente para desenvolvedores que buscam alto desempenho com flexibilidade.

## DataComPy

No campo da Engenharia de Dados, a tarefa de comparar DataFrames para identificar inconsistências é fundamental. O DataComPy é uma biblioteca Python robusta e fácil de usar para essa finalidade, compatível com Pandas, Spark e outras tecnologias. Com esta ferramenta, é possível realizar análises mais complexas dos dados, obtendo detalhes sobre as diferenças encontradas, definindo tolerâncias para colunas numéricas e ignorando diferenças conhecidas. Essa análise detalhada facilita a identificação da origem dos problemas e garante a qualidade dos dados.

## Pinia

O Pinia é uma solução moderna para o gerenciamento de estado das aplicações em Vue.js. Em comparação com o Vuex, o Pinia oferece uma API mais simples e intuitiva, baseada na Composition API, tornando o código mais organizado e fácil de ler. Além disso, o suporte nativo a TypeScript proporciona uma tipagem estática eficiente, diminuindo a probabilidade de erros e facilitando a manutenção do código.

## Ray

No contexto de Machine Learning, onde a necessidade de poder computacional cresce exponencialmente, o Ray se destaca como um framework potente para escalar aplicações Python e IA para clusters. Com ele, cientistas de dados podem executar tarefas de treinamento e inferência de modelos em larga escala, aproveitando o poder de processamento de várias máquinas. Sua integração com frameworks populares como PyTorch e TensorFlow, juntamente com um conjunto de bibliotecas otimizadas para IA, agregam para que o Ray se torne um bom framework para projetos de Machine Learning que requerem alto desempenho e escalabilidade.

## Conclusão

O Radar da Thoughtworks é como um mapa para quem trabalha com tecnologia e quer se manter atualizado e tomar decisões inteligentes. Ele não só mostra as tecnologias que estão em alta, mas também ajuda a entender quais delas vale a pena investir tempo e recursos, que precisam ser avaliadas com cuidado e quais podem trazer mais problemas do que soluções.

Com o Radar em mãos, as empresas podem desenvolver softwares e sistemas de forma mais rápida, eficiente e segura. Ele ajuda a escolher as ferramentas certas para cada projeto, otimizar o uso de recursos e evitar problemas no futuro. Em um mundo onde a tecnologia muda o tempo todo, ele é um guia valioso para quem quer inovar e se destacar.

