

# Resenha — Big Ball of Mud (1997), Brian Foote e Joseph Yoder

Aluno: Rafaela Oliveira Ribeiro

Matéria: Projeto de Software

O artigo “Big Ball of Mud” fala sobre um tipo de sistema muito comum: aquele código feito às pressas, com muitos remendos, difícil de entender e de manter. Os autores dão esse nome, que significa “grande bola de lama”, para descrever softwares que funcionam, mas não têm uma arquitetura clara. A ideia central é simples: por causa de prazos, custos e mudanças constantes, muitos projetos acabam ficando bagunçados, e isso acontece mais do que a gente imagina.

Os autores explicam por que esse cenário surge. Normalmente as equipes precisam entregar resultados logo, o escopo muda toda hora e nem sempre há tempo ou experiência para planejar bem. Assim, o caminho mais rápido vira o padrão. O sistema cresce, entra em produção e, como está “funcionando”, ninguém para para arrumar a casa. Com o tempo, qualquer ajuste vira um sofrimento, porque tudo está amarrado de um jeito confuso.

Para tornar isso mais claro, o texto descreve alguns padrões que aparecem na prática. O primeiro é o “Throwaway Code”: um código que seria só para teste, mas vira base do sistema. Depois vem o “Piecemeal Growth”, quando o software cresce aos pedaços, sem organização. Outro é o “Keep It Working”: a prioridade vira não quebrar o que já está no ar, então as mudanças são mínimas e muitas vezes superficiais. Há também o “Sweeping It Under the Rug”, que é esconder a bagunça atrás de uma fachada para que o resto do sistema não sofra tanto. Por fim, quando a situação fica insustentável, surge a “Reconstruction”, a reescrita geral.

O ponto forte do artigo é encarar a realidade sem moralismo. Em vez de só condenar o código feio, ele mostra as forças que nos empurram para lá e sugere saídas possíveis. A mensagem é: mantenha o sistema funcionando, mas vá melhorando aos poucos, isolando partes ruins e refatorando com frequência. O texto também usa metáforas simples, como a ideia de uma cidade que cresce sem plano, o que ajuda a visualizar o problema. Mesmo sendo de 1997, a discussão continua atual. Faltam temas modernos, como integração contínua e observabilidade, mas a base permanece verdadeira: se não cuidamos da arquitetura, a dívida técnica só aumenta.

Em resumo, “Big Ball of Mud” dá nome a um problema que quase todo mundo já viu e oferece um jeito pé no chão de lidar com ele. A grande lição é que “funcionar”

hoje não basta para sempre. É preciso reservar tempo para organizar o código, criar limites claros entre módulos e melhorar continuamente. Se isso não acontecer, a bola de lama cresce e cobrar esse preço mais tarde sai muito mais caro.