



Um dos “terrores” do technology lifecycle management: tecnologias com alta velocidade de atualizações e novas versões!

E por essa matéria da InfoWorld, algumas novidades da nova versão do Angular:

<https://www.infoworld.com/article/3681889/angular-16-to-improve-server-side-rendering.html>

De forma alguma eu tenho a intenção de criticar a tecnologia Angular. Acho que foi uma das grandes inovações em front-end da última década e conquistou o espaço que tem no mercado por ter muitas qualidades frente ao que existia.

Mas por outro lado, deixou bem claro as diferenças (muitas vezes não explicitadas)

entre as empresas tradicionais versus as digitais.

Nas tradicionais muitas vezes as soluções criadas são muito grandes (alto volume de telas) e pensadas para terem um ciclo de vida útil de muitos anos (historicamente medido as vezes em décadas).

Ao mesmo tempo, existem processo rígido de lifecycle management, onde se faz necessário garantir o nível de obsolescência baixo, ou seja, existe toda uma governança e processos estruturados para manter as soluções utilizando tecnologias em versões atuais e com o devido suporte dos seus respectivos vendor.

Aí surgem os dramas: como manter soluções gigantescas (eventualmente com milhares de telas) atualizadas em uma tecnologia que muda de versão 2 vezes ao ano e que uma dada versão perde suporte a cada ano?

Como criar arquiteturas que permitam a atualização faseada (imagina fazer big-bang de milhares de telas), convivendo partes da aplicação em uma versão e partes em outra?

Fora o drama de explicar as necessidades de budget recorrente para manter em dia esse processo de atualização constante.

Já as empresas digitais, em geral, tendem a renovar/evoluir suas camadas de interação com o cliente em um ritmo muito acelerado, de forma que a tecnologia acaba estando em geral muito mais atualizada de forma natural, pois ocorre via a própria renovação das jornadas.