

O Manifesto Reativo

Publicado em 16 de Setembro de 2014. (v2.0)

Organizações que trabalham em diferentes ramos, estão independentemente descobrindo padrões aleatórios para criarem sistemas semelhantes. Esses sistemas são mais robustos, mais resistentes, mais flexíveis e melhor posicionados para sustentar as demandas modernas.

Essas transformações estão acontecendo por causa dos requisitos que mudaram drasticamente nos últimos anos. Apenas alguns anos atrás, grandes aplicações tinham dezenas de servidores, demoravam segundos para responder, tinham manutenções que demoravam horas e lidavam apenas com gigabytes de dados. Hoje há aplicações em produção em todos os lugares, desde aplicativos móveis até aplicações na nuvem com clusters rodando milhares processadores multi-core. Geralmente os usuários esperam respostas em milisegundos e 100% de disponibilidade. Dados são mensurados em Petabytes. As demandas de hoje simplesmente não são mais atendidas pelas arquiteturas de ontem.

Nós acreditamos que é necessária uma abordagem coerente para arquitetura de sistemas, e acreditamos que todos os aspectos necessários já são reconhecidos individualmente: nós queremos sistemas Responsivos, Resilientes, Elásticos e Orientados a Mensagens. Nós chamamos isso de Sistemas Reativos.

Sistemas criados como Reativos são muito mais flexíveis, desacoplados e [escaláveis](#). Isso os torna mais fáceis de desenvolver e manter. São mais tolerantes a [falhas](#) e quando elas ocorrem são tratadas com elegância ao invés de desastre. Sistemas Reativos são responsivos, dando aos [usuários](#) feedbacks mais interativos.

Sistemas reativos são:

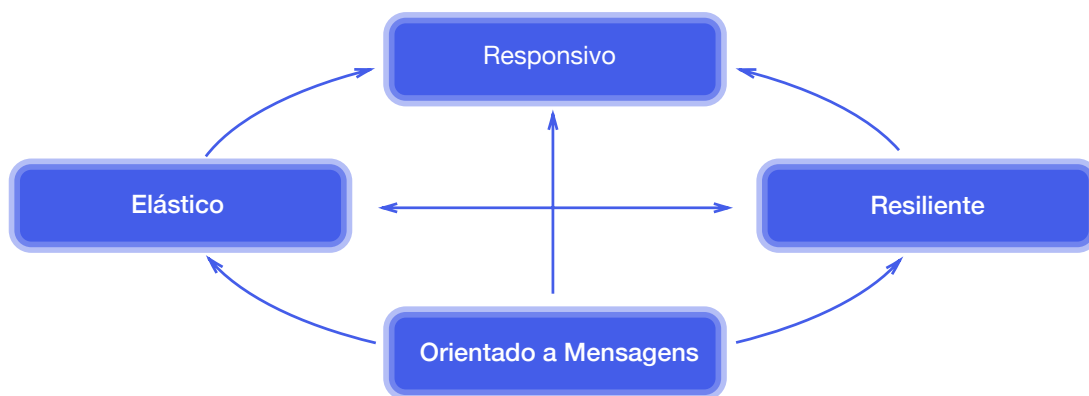
Responsivo: O [sistema](#) responde em um tempo hábil se possível. Ser responsivo é a pedra fundamental da usabilidade e utilidade, mas mais do que isso, responsividade significa que problemas podem ser detectados rapidamente e tratados com a máxima eficácia. Sistemas Responsivos são focados em fornecer tempos de resposta rápidos e consistentes, estabelecendo limites superiores de confiança para que eles entreguem qualidade de modo consistente. Esse comportamento consiste em simplificar o tratamento de erro, reforçar a confiança do usuário final e incentivar futuras interações.

Resiliente: O sistema continua respondendo em caso de [falha](#). Isto é aplicável não apenas para sistemas de missão crítica ou para alta disponibilidade — qualquer sistema que não é resiliente ficará fora do ar depois de uma falha. Resiliência é

alcançada por [replicação](#), contenção, [isolamento](#) e [delegação](#). A contingência a falhas é feita dentro de cada [componente](#), isolando-os uns dos outros e assim garantindo que partes do sistema podem falhar e se recuperar sem comprometer o sistema como um todo. A recuperação de cada componente é feita por outro componente (externo) e alta disponibilidade é garantida por replicação quando necessário. Os clientes de cada componente não são sobrecarregados com o tratamento de falhas.

Elástico: O sistema continua responsivo mesmo sob variações de demanda. Sistemas Reativos podem reagir a mudanças na taxa de entrada através do aumento ou diminuição dos [recursos](#) alocados para lidar com essas entradas. Isso requer projetos que não tenham pontos de contenção ou gargalos centrais, resultando na habilidade de dividir ou replicar componentes e distribuir a demanda entre eles. Sistemas Reativos suportam algoritmos de escalonagem preditivos, assim como Reativos, provendo métricas relevantes e em tempo real. Eles conseguem [elasticidade](#) com custo efetivo em hardware padrão e em plataformas de software.

Orientado a Mensagens: Sistemas Reativos usam [passagem de mensagens assíncronas](#) para estabelecer fronteiras entre os componentes e garantir baixo acoplamento, isolamento, [transparência na localização](#) e provêem meios para delegar o tratamento de [erros](#) através de mensagens. Empregar explicitamente a passagem de mensagens, modelar e monitorar as filas do sistema e aplicar [contrapressão](#) quando necessário, permite gerenciamento de demanda, elasticidade e controle de fluxo. Mensagens com transparência na localização como um meio de comunicação tornam possível o gerenciamento de falhas da mesma maneira seja em um cluster ou em um único host. Comunicação [não bloqueante](#) permite que destinatários consumam [recursos](#) quando ativos, evitando desgaste do sistema.



Grandes sistemas são compostos por pequenos serviços e portanto, dependem das propriedades Reativas de cada um deles. Isso significa que Sistemas Reativos usam princípios de projeto para que essas propriedades se apliquem em todos os níveis e escalas, tornando-os combináveis. Os maiores sistemas do mundo são arquitetados com base nessas propriedades e servem as necessidades de bilhões de pessoas diariamente. Está na hora de aplicar esses princípios conscientemente do início ao contrário de redescobri-los a cada hora.