



**Rational** software

## **Otimização da qualidade do software: equilíbrio entre transformações dos negócios e risco.**

*Michael Lundblad, Gerente de Programas, Software Rational,  
IBM Software Group*

*Moshe Cohen, Gerente de Ofertas, Software Rational,  
IBM Software Group*

---

## Conteúdo

---

- 2 **Introdução**
- 4 **Equilíbrio entre transformação dos negócios e risco.**
- 11 **Completando a visão**
- 13 **Soluções de gerenciamento da qualidade**
- 16 **Resumo**

***Nem sempre nos damos conta, mas interagimos com sistemas massivos de software todos os dias.***

### **Introdução**

Um estudo recente realizado pela IBM com CEOs mostrou que 66% esperam que suas organizações sejam inundadas por mudanças, amplamente causadas por inovação e transformação. Para manter o ritmo, as organizações de desenvolvimento de softwares precisam liberar softwares essenciais para os negócios em menos tempo, mas isso aumenta os riscos e, quase sempre, resulta em comprometimento da qualidade. Logo a pergunta é: *Como é possível poupar tempo e reduzir custos sem sacrificar a qualidade?*

Embora a aviação já tenha mais de 100 anos, a maioria de nós ainda fica admirado quando vemos um avião decolando ou aterrissando. O mais incrível é como o software de pilotagem automática controla e faz tudo com pouca intervenção do piloto. Pense na destreza do Joint Strike Fighter (JSF), também conhecido como F-35. Apontando para cima, ele pode parar no ar, manter sua posição e depois retomar seu vôo supersônico. Essa manobra também é controlada por software.

E os carros? Os veículos atuais contêm algo em torno de 100 a 150 computadores. São basicamente redes de computadores sobre rodas, que se comunicam continuamente entre si, dispõem de tecnologia global que fornece serviços como telemetria, posicionamento global, segurança interna e diagnósticos do sistema.

Todos os dias, interagimos direta e indiretamente com sistemas parrudos de software. O estoque dos nossos supermercados favoritos são controlados por software; nossas contas corrente e poupança são controladas por software; nossos financiamentos imobiliários, contas a pagar, prontuário médico e receitas médicas são controlados por software.

Esses avanços trazem praticidade e progresso, mas também representam riscos e gastos extras. Paul Ehrlich disse uma vez: “Errar é humano, mas, para bagunçar tudo de vez, você precisa de um computador”. Pense nos problemas relacionados a software relatados pela devtopics.com como um dentre os 20 piores dessa categoria de desastres.<sup>1</sup>

---

**Destaques**

---

***Em um ambiente de negócios no qual a alta qualidade deve ser alcançada com poucos recursos financeiros, o gerenciamento da qualidade dos softwares é crucial.***

Na “segunda-feira negra” (19 de outubro de 1987), a média industrial do Dow Jones despencou 508 pontos, perdendo 22,6% do seu valor total. O S&P 500 caiu 20,4%. A maior perda sofrida por Wall Street em um único dia foi causada por um problema de software. À medida que os investidores fugiam do mercado de ações como em um êxodo em massa, os programas de negociação por computador geravam uma avalanche de pedidos de venda, derrubando os sistemas e deixando os investidores efetivamente cegos e incapazes de negociar.

Em dezembro de 2008, o principal jornal de Israel (*Yediot Acharonot*) imprimiu uma edição com a manchete “O vírus que deixa os hospitais malucos”. O artigo não era sobre uma bactéria ou um vírus que deixa as pessoas doentes; era sobre um problema de software que associou os resultados de exames de sangue aos pacientes errados. O Ministério da Saúde de Israel informou que o problema surgiu quando os resultados dos exames foram transferidos eletronicamente dos laboratórios para os hospitais.

A entrega de softwares de qualidade nunca foi tão crucial para os negócios; na verdade, é crucial para a sobrevivência dos negócios. Na economia atual, os CIOs precisam equilibrar com cuidado a necessidade de transformação nos negócios com o gerenciamento dos riscos comerciais. Os requisitos estão sempre mudando, os cronogramas dos projetos sofrem atrasos, os custos são cuidadosamente analisados, mas a qualidade deve melhorar ou os negócios sucumbem. Essa tarefa aparentemente impossível está forçando as equipes de entrega de software a serem mais criativas, a examinar abordagens ágeis de desenvolvimento e a desafiar os métodos historicamente dominantes. Estão mudando sua visão de gerenciamento da qualidade, vendo-o como um esforço contínuo que requer trabalho em equipe, técnicas de automação, e relatórios precisos em tempo real ou a criação de um quadro geral das métricas para facilitar uma análise controlada da discussão.

Este artigo examina o equilíbrio entre transformação e riscos. Analisa as razões das mudanças no ambiente de desenvolvimento de softwares e discute as soluções que podem ajudar os negócios a melhorar o gerenciamento da qualidade para reduzir o tempo de comercialização, cortar custos e aprimorar a qualidade dos produtos.

---

**Destaques**

---

***Dos 20% que sobram do orçamento de TI após as despesas operacionais, três quartos são gastos procurando e resolvendo os problemas.***

***Manter o equilíbrio no “triângulo de ferro” é crucial para as empresas que querem inovação sem aumentar os riscos.***

**Equilíbrio entre transformação dos negócios e risco.**

A qualidade do software exige que as equipes encontrem o equilíbrio entre três dimensões: escopo (requisitos), custo e tempo (figura 1). Dependendo da pesquisa que você lê, aproximadamente 80% do orçamento de TI de uma organização é gasto com as operações. Dos 20% restantes, 70% a 80% são gastos procurando e resolvendo problemas nos aplicativos legados. Conseqüentemente, o financiamento de recursos para a entrega de qualidade do software é fixo, na melhor das hipóteses.

A experiência direta da IBM com organizações no mundo todo e os dados reunidos por pesquisadores que trabalham com centenas de empresas mostram que a maioria das empresas investem 25% ou mais do tempo e do custo do ciclo de vida do desenvolvimento com garantia da qualidade. Além disso, 30% dos custos de projeto para desenvolvimento de software estão associados a retrabalho e 70% desse valor estão relacionados a erros de requisitos.<sup>2</sup> Com as ferramentas atuais e os processos de desenvolvimento em cascata, a qualidade do software requer mais tempo e custo do que nunca. Para manter a competitividade, as empresas precisam encontrar maneiras de melhorar a qualidade e, ao mesmo tempo, encurtar o tempo de comercialização dos softwares cruciais para os negócios.



Figura 1: “O triângulo de ferro” — equilibrar a transformação dos negócios com os riscos.

---

### Destaques

---

***Quando os requisitos são mal definidos ou gerenciados, ou quando mudam sem um acompanhamento preciso, problemas sérios podem ocorrer.***

***As atualizações de software representam outra grande oportunidade de problemas nos códigos.***

Os requisitos nunca são fixos ... estão sempre mudando

A maioria dos problemas mais graves encontram seu caminho no software através dos requisitos: definição vaga, gerenciamento deficiente ou disseminação incompleta para todos os interessados. E esses problemas não surgem simplesmente no início do projeto; surgem durante todo o ciclo de vida do aplicativo. O desejo dos altos executivos seria que os requisitos fossem definidos e congelados, para que não pudessem ser alterados no futuro. Isso seguramente ajudaria a melhorar a qualidade dos softwares, mas a realidade atual exige que os requisitos continuem mudando. Eles refletem a velocidade dos negócios hoje em dia, a urgência de se manter competitivo e a necessidade de cumprir os regulamentos impostos à maioria dos nossos processos.

Vamos pensar em uma situação na qual o CIO percebe que as novas regulamentações exigirão que ele mude uma quantidade de aplicativos. Primeiro, o CIO e sua equipe precisam entender como essas mudanças vão afetar o software da empresa e ajustar suas atividades de teste para cuidar dessas mudanças. Enquanto isso, a extensão das mudanças, o tempo para testá-las manualmente e os custos associados convencem as empresas de que a melhora da automação, testes de regressão e processos iterativos ágeis são a melhor maneira de gerenciar os efeitos adversos inevitáveis das atualizações dos softwares.

Analisando o impacto da mudança nos requisitos em um mundo geograficamente distribuído.

Continuemos com o exemplo das mudanças regulamentares. A maioria das empresas lidaria com essas mudanças da forma mais natural e direta possível. Elas pediriam a cada equipe para analisar o impacto em seus aplicativos e agiriam de acordo. Nesse cenário, provavelmente acabaríamos com equipes diferentes realizando atividades similares simultaneamente nos mesmos produtos. Mas é comum que as empresas tenham vários projetos em andamento ao mesmo tempo em locais diferentes, internamente ou através de empresas contratadas, e todos precisam ser atualizados para refletir as mudanças regulamentares.

---

## Destaques

---

***A colaboração é essencial para se defender de possíveis problemas, porque permite que todos os membros da equipe se comuniquem durante a definição dos requisitos e as atualizações dos softwares.***

***Quando um aplicativo tem várias configurações, existe a tentação de pular algumas delas, mas isso aumenta o risco de entregar um software contaminado.***

Ninguém pode deixar de pensar que essa abordagem gera um grande desperdício, assim como aumenta os riscos. O que acontece se as diferentes equipes interpretarem os requisitos de maneiras distintas? E o que acontece se eles terminarem com casos de teste que refletem comportamentos diferentes quando deveriam ser idênticos? Lidar com vários projetos sem aumentar os riscos significativamente requer um processo de testes no qual um único conjunto de requisitos possa ser mapeado para múltiplos projetos. Requer a capacidade de definir casos de teste dos quais os scripts de texto possam ser gerados e/ou derivados. Requer a capacidade de compartilhar e reutilizar os casos de teste entre os projetos. Requer um plano de teste que, como os requisitos, não seja um documento fixo ou uma página de web hierárquica. É um bem ativo, dinâmico e sempre atual que permite a definição e o compartilhamento dos processos e estratégias de teste, relatórios em nível dos negócios de acordo com os objetivos de qualidade, e, também muito importante, que apóia atividades colaborativas, como, por exemplo, análises e aprovações.

Riscos associados aos ambientes de execução

Os softwares atuais são tão complexos que, em muitos casos, é simplesmente impossível garantir aplicativos sem erros. Em outros casos, mesmo que possível, não é financeiramente viável. Tome como exemplo uma situação na qual o portal com o cliente precisa ser testado. O primeiro passo é pedir à equipe de testes para desenvolver casos de teste, alguns manuais e outros automatizados. Dependendo da complexidade do aplicativo e do ambiente pretendido, o ciclo de testes pode levar de dias a semanas ou até mesmo meses. Na verdade, vamos supor que o portal oferece suporte apenas a dois idiomas e quatro navegadores ou versões diferentes, e é executado em três bancos de dados e um servidor de aplicativos. Esses poucos componentes por si só ainda requerem que o ciclo de teste seja repetido 24 vezes (2 x 4 x 3 x 1). Portanto, o que levava uma hora para ser testado agora leva 24. O que levava uma semana de 40 horas agora leva 24 semanas, ou seja, quase meio ano. À medida que crescem os números de configurações que precisam ser testadas, é tentador eliminar o teste de algumas delas, aumentando com isso o risco de entregar softwares defeituosos.

---

### Destaques

---

***A IBM descobriu que testar um pequeno subconjunto de combinações de configurações pode ajudar os negócios a detectar uma porcentagem muito alta de defeitos com mais rapidez.***

A IBM pesquisou uma maneira promissora de reduzir significativamente os riscos associados a um grande número de ambientes de execução. Os resultados mostraram que existe um pequeno subconjunto de combinações que pode detectar uma porcentagem muito alta de defeitos com mais rapidez. Basicamente, ele acelera a detecção de defeitos. Como mostrado na figura 2, os testes em múltiplos ambientes de execução seguem a curva “lenta” típica, na qual os defeitos são detectados sem uma prioridade especial. A detecção de defeitos com o tempo fica, em grande parte, na zona aleatória, e os cronogramas são comprometidos. No entanto, a aplicação de otimizações inteligentes traz a detecção de defeitos com o tempo para a zona otimizada, na qual os defeitos são detectados muito mais cedo no ciclo de testes, poupando tempo e reduzindo significativamente o risco de entregar softwares contaminados.

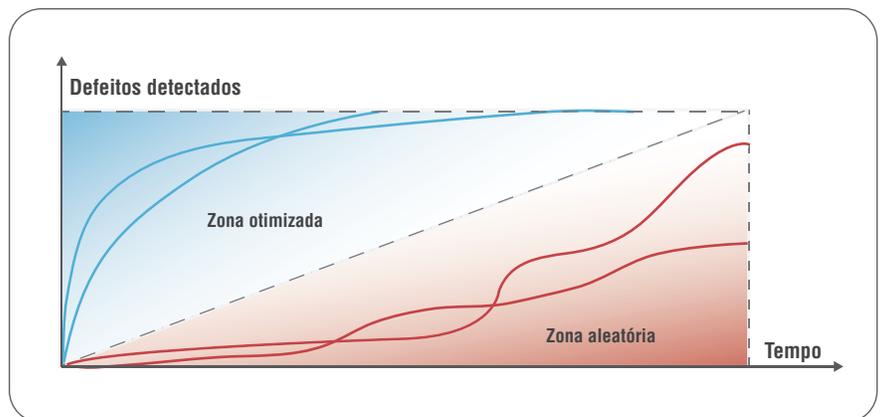


Figura 2: Acelerar a descoberta de defeitos com otimizações inteligentes

---

### Destaques

---

***A incorporação do gerenciamento da qualidade logo no início do ciclo de vida do desenvolvimento de softwares pode ajudar as organizações a evitar o alto custo de resolver os problemas mais tarde no jogo.***

#### Mudança de paradigma no gerenciamento da qualidade

Para enfrentar esse desafio na entrega de software de qualidade, as organizações devem se engajar em processos colaborativos propiciados pela automação e acompanhados por controle de projeto viável durante o ciclo de vida de fornecimento. Os testes de garantia da qualidade (QA, quality assurance) tradicionais simplesmente confirmam que o software em desenvolvimento, antes da implementação, atende às expectativas do usuário final em termos de funcionalidade, disponibilidade e desempenho. Encontrar os defeitos na fase de testes de QA é muito mais caro e consome muito mais tempo do que encontrar os defeitos no início do ciclo de vida do desenvolvimento do aplicativo (figura 3).

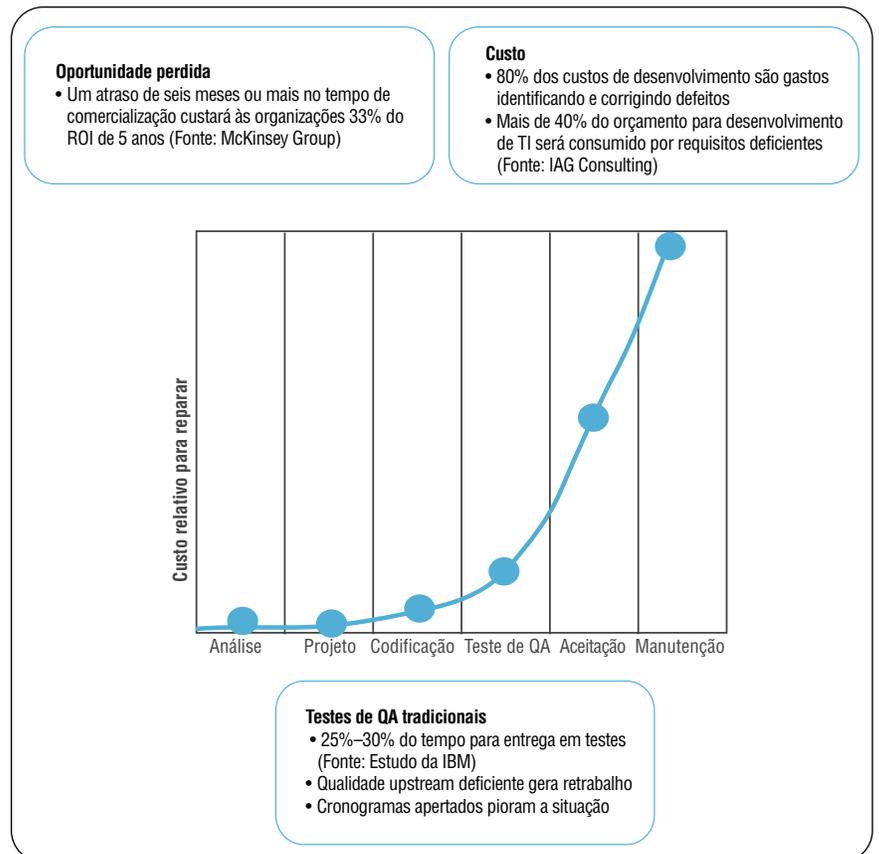


Figura 3: As oportunidades de otimizar a qualidade e o tempo de comercialização existem em todo o ciclo de vida do desenvolvimento de aplicativos.

---

### Destaques

---

***A colaboração enfrenta três barreiras principais: dispersão geográfica das equipes, infra-estruturas incompatíveis e desalinhamento das capacidades dos funcionários.***

O gerenciamento da qualidade se tornou uma prática de equipe, mas a colaboração enfrenta muitas barreiras. As *barreiras geográficas* bloqueiam as comunicações entre as equipes em vários fusos horários e criam falhas de processo que resultam em retrabalho. As *barreiras de infra-estrutura* criam problemas de processos, ferramentas e integração dos dados, que impedem as equipes de receber informações em um prazo razoável, o que resulta em atrasos nos cronogramas. Por fim, as *barreiras organizacionais* introduzem problemas com experiência em domínios, controle de projetos deficiente e dilemas da terceirização, como a segurança da propriedade intelectual.

Como as principais empresas do mercado estão melhorando a qualidade dos softwares? Uma análise dos Serviços Globais da IBM<sup>3</sup> de 846 projetos em vários clientes revelou um valor diferenciado significativo com abordagens colaborativas e automatizadas. Processos de teste abrangentes, tecnologias integradas de ciclo de vida completo, caso de teste e reutilização de scripts baseados na indústria, e análise de defeitos e processos de gerenciamento da qualidade avançados mostraram as seguintes faixas de aprimoramento:

- *Melhoras na qualidade de 30% a 70%*
- *Melhoras no tempo de ciclo de 20% a 50%*
- *Reduções de custos de 15% a 60%*

	Desenvolvimento de soluções de testes reproduzíveis do setor			Análise de defeitos avançada	Desenvolvimento de procedimentos de testes reproduzíveis aplicáveis a futuros projetos		Integração de processos completos		
Ativo	Casos de testes copiados	Roteiros manuais copiados	Reutilização de roteiros manuais	Impedir e bloquear defeitos duplicados	Documentação da linha de base e migração	Artefatos da linha de base	Alavancar a reutilização de componentes	Atualizações de dinâmicas de ativos de testes	Total
Quantidade	343	350	1.393	905	1.365	2.023	1.029	2.227	9.635
Horas economizadas	167	175	696	1.755*	683	1.011	515	557	5.558
Valor	\$16.690	\$17.514	\$69.633	\$175.452	\$68.254	\$101.125	\$51.459	\$55.673	
<b>Total</b>			<b>\$103.387</b>	<b>\$175.452</b>		<b>\$169.379</b>		<b>\$107.132</b>	<b>\$555.799</b>

\* As horas poupadas supõem uma média de quatro horas para detectar a duplicação. Na verdade, freqüentemente leva muito mais.

Tabela 1: Resultados diferenciados dos valores de negócios com uma abordagem automatizada, guiada por processos e colaborativa ao gerenciamento da qualidade.

A tabela 1 resume algumas das descobertas deste estudo, que mostra como 846 projetos desses clientes pouparam em média mais de meio milhão de dólares por projeto.

Onde essas organizações encontraram economias? Em seus processos de gerenciamento da qualidade.

- **Desenvolver soluções de teste da indústria reproduzíveis.** Parece razoável que, em qualquer setor, os casos de teste e os scripts manuais se tornem bastante similares. Isso significa que copiar e reutilizar esses casos de teste e scripts comuns poderiam representar economias substanciais. Neste estudo, a economia média por projeto foi de mais de US\$ 100.000 (16.690 + 17.514 + 69.633).
- **Realizar análise de defeitos avançada.** Não é fácil detectar e evitar defeitos duplicados, mas é muito importante detectá-los cedo. Se não fizer isso, você corre o risco de ter múltiplas equipes trabalhando no mesmo defeito sem saber, produzindo com freqüência mais defeitos. A detecção automatizada de defeitos duplicados não apenas melhora a qualidade, como também reduz os riscos e os custos. Neste estudo, supondo uma média de quatro horas para detectar a duplicação, a economia média por projeto foi de US\$ 175.452.

**As organizações podem poupar tempo e dinheiro prestando mais atenção ao gerenciamento da qualidade em todo o ciclo de vida de desenvolvimento e entrega de softwares.**

---

### Destaques

---

- **Desenvolver procedimentos de teste reproduzíveis que possam ser reaplicados em projetos futuros.** Desenvolver novos artefatos para cada projeto, em vez de reutilizar os artefatos disponíveis, consome tempo e é desnecessário. Padronizar com procedimentos e técnicas de teste reproduzíveis pode representar uma economia de tempo considerável. A economia média por projeto foi de cerca de US\$ 170.000 (68.254 + 101.125).
- **Integrar processos completos.** A integração do gerenciamento dos requisitos com as técnicas de gerenciamento da qualidade fornece acompanhamento bidirecional total dos requisitos, casos de teste e resultados dos testes. Permite também várias análises de impacto, principalmente quando os requisitos mudam durante o desenvolvimento. Isso é essencial para obter vantagens competitivas aumentando a qualidade e reduzindo os custos. A economia média por projeto foi de mais de US\$ 100.000 (51.459 + 55.673).

Em resumo, uma abordagem guiada por processos, colaborativa e automatizada ao gerenciamento da qualidade não apenas permitiu estratégias de mitigação dos riscos e mais qualidade, como também poupou, em média, mais de US\$ 500.000 por projeto!

#### Completando a visão

Para cumprir a promessa de inovação e crescimento — por exemplo, agilidade comercial através de uma arquitetura orientada a serviços (SOA) e desenvolvimento de softwares baseado em componentes em todas as equipes geograficamente distribuídas  as organizações devem estabelecer uma estratégia de gerenciamento da qualidade unificada e rápida. As equipes de entrega de softwares e sistemas podem ajudar nesse esforço examinando três áreas principais.

**Uma estratégia de gerenciamento da qualidade forte pode ajudar uma organização a se posicionar para inovação e crescimento.**

---

**Destaques**

---

***As organizações devem estabelecer uma estratégia de gerenciamento da qualidade que envolva colaboração, automação e controle viável.***

#### Colaboração

As equipes devem se manter em sincronia com processos otimizados e dinâmicos e fluxos de trabalho baseados em atividades. Os requisitos de negócios sempre em mudança devem fluir pelo processo de qualidade para garantir que os casos de teste estejam atualizados e que os desenvolvedores entendam os requisitos mais recentes. Os gerentes de QA e projeto precisam ter certeza de que suas equipes estão trabalhando nas tarefas mais prioritárias. O planejamento de testes deve ser contínuo e envolver uma abordagem orientada a objetivos com critérios de entrada/saída e configurações ambientais priorizadas.

#### Automação

O desenvolvimento de softwares já viu atividades profissionais individuais, como testes funcionais e de desempenho, serem automatizadas. E os engenheiros de build atuais usam scripts e ferramentas de build que poupam tempo. Mas é preciso fazer mais para automatizar o processo e as etapas entre as funções para melhorar as eficiências organizacionais, poupar dinheiro e acelerar o tempo de comercialização (p.ex., reutilização de casos de teste e scripts de teste em todas as linhas de negócios, análise/fornecimento automatizado de laboratórios de teste e conversão automatizada de modelos de caso de uso em casos de teste).

#### Controle viável

Os resultados de todos os tipos de testes — incluindo testes de unidade, funcionais, de integração e escalabilidade — devem estar disponíveis imediatamente para geração de relatórios e análise de tendências. A integração das informações técnicas de projeto aos analíticos de negócios apóiam a influência dos tomadores de decisão de nível mais alto na alocação e utilização dos recursos para alinhar a TI aos negócios. Os recursos de gerenciamento de testes e planejamento devem estar fortemente integrados à análise de requisitos e aos processos de definição. As equipes de testes envolvidas no planejamento antecipado de casos de testes devem ter acesso em tempo real aos requisitos

---

**Destaques**

---

***Diretrizes de processos, melhores práticas e ferramentas de equipe integradas são elementos essenciais nas soluções de gerenciamento da qualidade atuais.***

comerciais e funcionais de projeto, casos de uso e contratos de níveis de serviços (SLAs, service-level agreements). A integração dos dados de status de projeto permite aos analistas de negócios comparar as mudanças de projeto com os objetivos e as restrições comerciais. A avaliação desses dados durante cada fase permite que as equipes de projeto façam ajustes e avaliem os riscos prioritários. O resultado geral propicia mais controle de projeto e, quando surgem problemas no projeto, uma avaliação dos riscos de negócios mais precisa. As informações certas na hora certa, focadas e filtradas para análise de decisões, permitem uma produção de softwares controlada e viável.

**Soluções de gerenciamento da qualidade**

As soluções de gerenciamento da qualidade atuais fornecem diretrizes de processo para entrega de software, melhores práticas e ferramentas de equipe integradas para ajudar as organizações a alcançar melhorias em termos de eficiência de custos, métricas de qualidade e obtenção de valor em projetos de negócios críticos.

Como demonstrado acima, a colaboração é essencial. As pessoas podem colaborar usando e-mail ou compartilhamento aprimorado de arquivos, mas esses métodos são tão cansativos que nem fazem parte do processo central de entrega de software. Várias organizações desenvolveram ferramentas colaborativas de gerenciamento do ciclo de vida do aplicativo (ALM, application lifecycle management) que tornam o progresso das equipes transparente e controlam o trabalho em equipe. A plataforma Jazz™ da IBM inclui recursos

---

### Destaques

---

***As ferramentas de gerenciamento da qualidade fornecem recursos de colaboração, automação e geração de relatórios para ajudar as empresas a entregar, cada vez mais, produtos de qualidade superior.***

de ALM para dar suporte a gerenciamento de requisitos, atividades de desenvolvimento e gerenciamento da qualidade, além de ajudar as equipes a superar as barreiras geográficas, de infra-estrutura e organizacionais que dificultam a colaboração. Além disso, centros baseados na web permitem que as pessoas trabalhem juntas para entregar qualidade de software duradoura, como um bem estratégico dos negócios. As novas soluções permitem aos usuários:

- ***Colaborar*** em todas as equipes de negócios, desenvolvimento e testes com fluxos de trabalho dinâmicos baseados em processos e atividades para o planejamento e a execução de testes.
- ***Automatizar*** os processos de ciclo de vida que utilizam muita mão-de-obra e detectar os problemas de qualidade logo no início, reduzindo o tempo de comercialização, cortando custos e mitigando os riscos de negócios.
- ***Registrar*** métricas priorizadas personalizadas para indivíduos e equipes, facilitando a visibilidade e permitindo que os tomadores de decisão atuem com mais confiança.
- ***Entregar*** mais previsibilidade, mapeando padrões de implantação bem-sucedidos nos indicadores-chave de desempenho (KPIs) operacionais.

As empresas que aproveitarem as vantagens das novas ferramentas de definição/gerenciamento de requisitos poderão:

- ***Alinhar*** os projetos de desenvolvimento aos objetivos de negócios para reduzir os riscos de falhas de projeto que custam bilhões de dólares anualmente às organizações.
- ***Permitir*** melhores resultados e validação de requisitos entre os especialistas de negócios e técnicos usando técnicas visuais e colaborativas comprovadas (p.ex., esboços de processos de negócios, esboços e esquemas de interface do usuário, e casos de uso).
- ***Gerenciar*** com mais eficiência as mudanças nos requisitos e no escopo dos projetos (p.ex., análise de impacto).
- ***Melhorar*** o tempo de comercialização e o retorno sobre o investimento dos projetos reduzindo o retrabalho devido a requisitos deficientes ou à ausência de requisitos.

---

---

**Destaques**

---

---

***As ferramentas de ALM ajudam a produzir e a garantir a qualidade por meio de modelagem arquitetônica, desenvolvimento, testes funcionais e de sistemas, testes de validação de build, testes manuais, testes de carga e escalabilidade, e suporte a aplicativos de produção.***

Como parte da estratégia de qualidade de ALM, as ferramentas de produção de software, como as descritas na tabela 2, podem ajudar a garantir a qualidade e o desempenho dos softwares cruciais para os negócios.

Ferramentas	Descrição
Modelagem arquitetônica	<ul style="list-style-type: none"><li>• Valida as regras definidas pelos usuários que representam restrições arquitetônicas.</li><li>• Detecta automaticamente padrões de projeto e estruturas importantes baseadas em objetos.</li><li>• Detecta antipadrões estruturais (como, por exemplo, nós, centros de distribuição e interseções) que degradam o desempenho.</li><li>• Redirecionam os nós automaticamente por meio de correções rápidas.</li></ul>
Desenvolvimento	<ul style="list-style-type: none"><li>• Ajuda os desenvolvedores a detectar corrupção de memória e vazamentos, perfil de desempenho e cobertura de código.</li></ul>
Testes de validação de build automatizados	<ul style="list-style-type: none"><li>• Evita a implantação de falhas de build no laboratório de testes ou no ambiente de testes de sistema.</li><li>• Permite que os desenvolvedores aproveitem os ciclos do horário não comercial para testar a estabilidade e a funcionalidade do aplicativo.</li></ul>
Testes manuais	<ul style="list-style-type: none"><li>• Promove melhores práticas como modularidade e capacidade de reutilização dos testes para fazer a transição das equipes dos testes manuais para os automatizados.</li></ul>
Testes funcionais e de sistemas	<ul style="list-style-type: none"><li>• Permite às equipes criar testes que verificam de maneira manual ou automática os erros de regressão e funcionais.</li><li>• Diminui os ciclos de automação de testes para melhorar a qualidade através de uma cobertura de testes mais ampla e mais profunda.</li><li>• Incentiva testes mais precisos, confiáveis e reproduzíveis.</li></ul>
Testes de carga e escalabilidade	<ul style="list-style-type: none"><li>• Determina os limites de carga e escalabilidade para tecnologias e aplicativos como Plataforma Java™, Enterprise Edition (Java EE); Web (particularmente portais); SOA; Siebel; Oracle e SAP.</li><li>• Garante que um aplicativo de software possa se expandir e atuar para atender aos SLAs e às expectativas dos usuários.</li><li>• Permite aos testadores apontar a origem dos gargalos de desempenho, para que seja possível navegar até o código fonte sem perder tempo tendo de passar por múltiplos níveis de código.</li><li>• Permite um retorno melhor sobre os investimentos em hardware realizando testes pré-implantação de planejamento de capacidade que dimensionam os recursos de servidor necessários para alcançar o desempenho e os resultados desejados.</li></ul>
Soluções de suporte a aplicativos de produção	<ul style="list-style-type: none"><li>• Permite a colaboração entre as equipes de operações e desenvolvimento para apoio aos aplicativos, isolamento de problemas e reparo de circuito fechado.</li><li>• Captura informações de registro de transações e acompanhamento, assim como dados de recursos de sistema estendidos para uma determinação de problemas mais granular, reduzindo assim o tempo de resposta aos diagnósticos de suporte aos aplicativos.</li><li>• Permite restauração mais rápida dos níveis de serviços para o negócio.</li></ul>

Tabela 2: As ferramentas de ALM contribuem para a entrega de ciclo de vida da qualidade.



## Resumo

Nesta década, vimos uma colisão de três vias entre a busca da transformações dos negócios através de inovação, a necessidade de aprimorar a qualidade do software para gerenciar os riscos de negócios e a demanda por redução dos custos para sobrevivência econômica.

A mudança do setor para um trabalho em equipe colaborativo entre as equipes geograficamente distribuídas, juntamente com processos e geração de relatórios automatizados para dar suporte a um controle viável, está mudando a cara da entrega de softwares de qualidade.

Gerenciamento da qualidade não significa mais equipes de QA com ferramentas de testes automatizadas dando suporte à entrega de softwares tradicional; também não significa processos de controle do gerenciamento da qualidade que atrasam a produção ou simplesmente testam e reparam os defeitos. Gerenciamento da qualidade significa inserir qualidade em um ciclo de desenvolvimento iterativo e em um programa completo de ajuste do software, apoiado por ferramentas e dados integrados e métricas que podem ser acompanhadas. Um gerenciamento da qualidade eficaz torna o processo de produção mais gerenciável e menos doloroso, e ajuda a gerar confiança nas equipes de operações. Basicamente, quebra o triângulo de ferro mencionado anteriormente neste artigo, porque permite aos gerentes otimizar o escopo, os custos e os prazos, além de melhorar a qualidade.

Promover o gerenciamento da qualidade em todo o ciclo de vida e detectar os defeitos logo no início do processo reduz os custos e melhora a credibilidade. Comunicação constante através de um conjunto de requisitos comum e bem entendido — e respostas rápidas às mudanças dos requisitos — pode inserir qualidade no desenvolvimento de software logo de saída. Além disso, um controle cuidadoso dos processos de build e execução de testes para corrigir o curso e alocar recursos pode ajudar as organizações a se tornarem mais flexíveis, cuidar das regulamentações de conformidade, reagir com mais rapidez às mudanças do mercado e, finalmente, promover o crescimento dos negócios. Nesta economia implacável, o sucesso ou o fracasso freqüentemente depende de uma coisa: de quem é o produto de melhor qualidade.

## Para obter mais informações

Para saber mais sobre as soluções da IBM de gerenciamento da qualidade, entre em contato com o seu representante IBM ou acesse:

[ibm.com/software/rational/offerings/quality](http://ibm.com/software/rational/offerings/quality)

Agradecimentos especiais a Ron French da IBM Global Business Services por compartilhar descobertas importantes do seu trabalho e do trabalho de sua equipe.

© Copyright IBM Corporation 2009

IBM Corporation  
Grupo de software  
Route 100  
Somers, NY 10589  
EUA

Produzido nos Estados Unidos da América  
Março de 2009  
Todos os direitos reservados

IBM, o logotipo IBM, ibm.com e Rational são marcas comerciais ou registradas da International Business Machines Corporation nos Estados Unidos e/ou em outros países. Se estes ou outros termos registrados da IBM estiverem marcados em sua primeira ocorrência nesta informação com o símbolo de marca registrada (® ou ™), esses símbolos indicam marcas comerciais registradas ou de lei comum de propriedade da IBM no momento em que esta informação foi publicada. Essas marcas comerciais também podem ser marcas registradas ou de direito comum em outros países. Uma lista atual das marcas comerciais da IBM está disponível na web em "Copyright and trademark information" (Informações sobre direitos autorais e marcas comerciais) em [ibm.com/legal/copytrade.shtml](http://ibm.com/legal/copytrade.shtml)

Java e todas as marcas registradas baseadas em Java são marcas registradas da Sun Microsystems, Inc. nos Estados Unidos e/ou em outros países.

Outras empresas, nomes de serviços e produtos podem ser marcas comerciais ou registradas de terceiros.

As referências feitas nesta publicação a produtos ou serviços IBM não pressupõem uma intenção da IBM de torná-las disponíveis em todos os países nos quais opera.

As informações contidas nesta documentação foram fornecidas com o objetivo único de informar. Apesar dos esforços para verificar a integridade e a precisão das informações fornecidas, estas foram fornecidas "na forma em que se encontram", sem quaisquer garantias, expressas ou implícitas. Além disso, estas informações se baseiam nos planos e estratégias de produtos atuais da IBM, que estão sujeitos a mudança sem aviso prévio. A IBM não deverá ser responsabilizada por quaisquer danos causados pela utilização desta documentação ou de qualquer outra documentação, ou danos de qualquer maneira relativos às mesmas. Nenhum conteúdo desta documentação tem a intenção de, nem deverá gerar quaisquer garantias ou representações da IBM (ou de seus fornecedores ou licenciadores), ou alterar os termos e condições do contrato de licença que governa a utilização do software IBM em questão.

- 1 <http://www.devtopics.com/20-famous-software-disasters>
- 2 Walker Royce, *Software Project Management: A Unified Framework (Gerenciamento de projetos de software: uma estrutura unificada)*, Addison-Wesley/Professional, Indianápolis, 1998.
- 3 Serviços Globais IBM, SEANT (Systems Engineering Architecture & Test) Programa de ReUtilização, 2007.