

UM GUIA DA ABACCUS SOBRE BRMS

Onde mora a regra de negócio

Por que a parte do seu software que mais muda está
no lugar errado, e quanto isso custa.

Este é um guia de categoria, não de produto. Ele trata de um problema que atravessa setores: a regra de negócio, a parte do software que mais muda, quase nunca tem um lugar próprio. Em vários momentos vamos dizer abertamente quando um BRMS não é a resposta. Isso faz parte de uma conversa honesta.

Sumário

00 **A premissa**

01 **Onde a regra mora hoje**

02 **O que é um BRMS de verdade**

03 **O custo invisível**

04 **Explicabilidade e rastreabilidade**

05 **IA e BRMS: a arquitetura híbrida**

06 **Quando não usar um BRMS**

07 **Como começar sem big bang**

- **Na prática: cinco setores, o mesmo padrão**

- **Para ir além: site, comparativos e IA**

A parte do software que mais muda

Toda empresa que cresce descobre, mais cedo ou mais tarde, a mesma coisa: a parte do sistema que mais muda não é a interface, nem o banco de dados. É a regra de negócio.

O critério que aprova ou recusa um cliente. A faixa que define um preço. A condição que libera uma comissão, autoriza um sinistro ou classifica um risco na aceitação. São essas decisões que mudam toda semana, por pressão comercial, regulatória ou competitiva.

O paradoxo é que essa parte tão volátil quase nunca tem um lugar próprio. Ela se espalha. Vive chumbada no código, dentro de um fluxo de uma ferramenta de SaaS, num notebook do data lake, numa planilha que alguém mantém de cabeça, ou pior, na cabeça de uma ou duas pessoas. Cada mudança vira um pedido para a fila da engenharia. Cada auditoria vira uma escavação.

Acompanhamos esse padrão se repetir em seguradoras, financeiras, varejistas, operações de logística e áreas comerciais. O setor muda, o sintoma não.

A PERGUNTA DESTE GUIA

Onde a regra de negócio mora hoje, por que esse lugar costuma ser o errado, quanto isso custa de verdade, e como dar a ela um endereço próprio: governado, explicável e mudável por quem entende do negócio, sem depender de um novo deploy.

Onde a regra mora hoje

Cada plataforma serve para uma coisa. O problema é quando a empresa usa ela para a coisa que ela não foi feita para fazer.

Quando uma decisão importante precisa existir, ela é colocada no lugar mais à mão naquele momento. Faz sentido no curto prazo. O custo aparece depois, quando o lugar provisório virou definitivo e ninguém mais quer mexer.

Vemos a mesma regra crítica aparecer em cinco endereços típicos:

- **No código da aplicação**, na forma de condicionais que se multiplicaram ao longo dos anos. Mudar um critério exige um desenvolvedor, um pull request, testes e um deploy.
- **Dentro de um fluxo de uma ferramenta de SaaS** (um Flow de CRM, uma automação de service desk). A ferramenta é ótima no que faz, mas a lógica de decisão fica refém de quem domina aquela ferramenta específica.
- **Num notebook ou view do data lake**, onde um score é calculado. O score é excelente. A decisão sobre o que fazer com ele acaba grudada no pipeline de dados.
- **Num painel de BI**, que mostra o número mas não decide nada, e ainda assim vira a fonte informal da regra.
- **Numa planilha paralela**, que ninguém versiona e todo mundo teme.

Nenhum desses lugares foi feito para abrigar uma regra que muda toda semana, precisa ser explicada para um auditor e deveria ser ajustada por quem é do negócio.

VOCÊ RECONHECE SUA OPERAÇÃO SE

mudar um critério de aprovação leva semanas e passa pela fila do TI; ninguém sabe ao certo qual versão da regra está em produção; responder "por que esse caso foi recusado?" exige abrir o código; e uma ou duas pessoas são insubstituíveis porque só elas entendem a lógica.

Esses sintomas não são falha de ninguém. São o resultado natural de tratar a parte mais volátil do sistema como se fosse a mais estável.

O que é um BRMS de verdade

BRMS é a sigla para **Business Rules Management System**, sistema de gestão de regras de negócio. Na prática, é um motor que tira a lógica de decisão de dentro do código e a coloca numa camada própria, central, versionada e desacoplada do ciclo de desenvolvimento.

A diferença não é cosmética. Quando a regra vive num motor gerenciado:

- ela é escrita e alterada em linguagem de negócio (low-code), sem exigir um deploy a cada ajuste;
- cada versão fica registrada, com quem alterou, quando e por quê;
- a mesma regra pode ser consumida por qualquer sistema via API, em vez de ser reescrita em cada um;
- toda decisão deixa rastro, o que torna a auditoria uma consulta, não uma escavação.

O que um BRMS não é

É aqui que mora boa parte da confusão. Um BRMS não é um CRM (o CRM gerencia o relacionamento, não a regra). Não é uma ferramenta de BI (o BI mostra o número, não toma a decisão). Não é um data lake (o data lake calcula o score, a decisão sobre o score é outra camada). E não é uma pasta de condicionais no código, mesmo que esse seja, hoje, o lugar onde a sua regra está.

A pergunta certa não é "qual ferramenta tem regras?". Quase todas têm. É "onde a regra de negócio é dona da casa, em vez de inquilina?"

O custo invisível

O custo do open source não está na licença. Está em tudo que vem em volta dele.

Existem boas plataformas de regras no mercado. O Drools é maduro, poderoso e gratuito na licença. FICO Blaze Advisor e IBM ODM são robustos e provados em ambientes críticos. Reconhecer isso é o ponto de partida honesto. O que costuma passar despercebido não é a qualidade dessas opções, é o custo total de mantê-las.

No open source, o custo não está na licença, está no que vem em volta: o time especializado, a infraestrutura, o versionamento que você mesmo constrói, a trilha de auditoria que não vem pronta. O preço de aquisição é zero. O custo de propriedade não é.

Nas plataformas licenciadas há um fator mais silencioso: o modelo de cobrança. A maioria cobra por transação. O problema é o incentivo que isso cria. Quanto mais a sua operação cresce e mais decisões você automatiza, maior fica a conta. Você é penalizado por crescer.

A DIFERENÇA ESTRUTURAL DA ABACCUS

A nossa cobrança é uma mensalidade por motor de regras, não por transação. Dez mil decisões por mês ou cem milhões: o custo do motor é o mesmo. Em volume enterprise, a diferença entre os dois modelos vira uma decisão de arquitetura financeira.



Ilustrativo. O ponto não são os valores absolutos, é o formato das curvas: uma cresce com o volume, a outra não.

Explicabilidade e rastreabilidade

Toda decisão automatizada carrega uma pergunta que, mais cedo ou mais tarde, alguém vai fazer: por que essa decisão foi tomada assim? Pode ser um auditor, um regulador, um cliente que recorreu, ou a própria diretoria revisando um resultado.

Quando a regra vive no código, responder é um projeto. Alguém precisa encontrar a versão que estava em produção naquela data, ler a lógica e reconstruir o raciocínio. Quando a regra vive num motor gerenciado, responder é uma consulta: cada decisão fica registrada com a versão da regra que a gerou, os dados de entrada e o resultado.

Vale separar três capacidades técnicas concretas, que são propriedades da arquitetura e não selos:

- **Explicabilidade:** saber por que a decisão foi aquela.
- **Rastreabilidade:** reconstruir o caminho que levou até ela.
- **Trilha de auditoria:** o registro de quem mudou o quê e quando.

Estão entre as razões pelas quais áreas de risco e controle preferem regra externalizada a regra escondida no sistema.

UMA DISTINÇÃO IMPORTANTE

Setores como seguros e finanças operam sob exigências de órgãos como SUSEP e Bacen e sob a LGPD. Esse contexto regulatório aumenta o valor de decisões explicáveis. Mas é preciso separar as coisas: explicabilidade e audit trail são capacidades de engenharia que você pode verificar. Conformidade é um processo da sua empresa, do qual essas capacidades são insumo, não substituto.

A arquitetura híbrida

O agente raciocina. O motor decide. Quem confunde os dois paga o preço na auditoria.

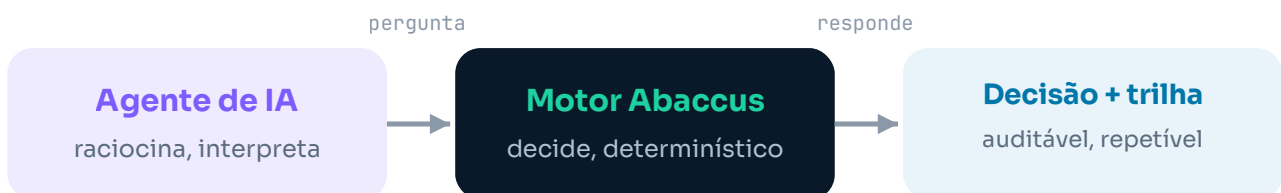
Os próximos anos vão colocar agentes de IA dentro de praticamente todo departamento e todo produto. Atendimento, operações, crédito, precificação, suporte. Isso multiplica o número de decisões automatizadas numa empresa. E é exatamente por isso que a camada de regras fica mais importante, não menos.

Um agente de IA é, por natureza, probabilístico. Ele raciocina, interpreta linguagem, lida com ambiguidade. É ótimo nisso. Mas uma decisão de negócio crítica (aprovar, recusar, precificar, pagar) precisa ser determinística, repetível e explicável. A mesma entrada tem que produzir a mesma saída, sempre, e você precisa conseguir mostrar por quê.

O ANTIPADRÃO

Colocar a regra dentro do prompt do agente, ou esperar que o modelo "lembre" da política, é trocar uma decisão auditável por uma aposta. Funciona até o dia em que não funciona, e nesse dia não há trilha de auditoria que explique o que aconteceu.

A arquitetura que faz sentido separa os papéis. O agente raciocina e conversa. Quando chega a hora de decidir, ele consulta o motor de regras, que responde de forma determinística e deixa rastro. Em termos práticos, o BRMS passa a ser uma ferramenta que o agente consome, no vocabulário atual, exposta por um protocolo como o MCP. Vale como regra geral: quanto mais IA a sua empresa adota, mais necessário fica um lugar determinístico onde as decisões realmente acontecem.



Quando não usar um BRMS

Filtrar cliente parece contraintuitivo para quem vende. Não é. Honestidade economiza tempo dos dois lados.

Seria fácil terminar este guia dizendo que toda empresa precisa de um BRMS agora. Não é verdade, e dizer a verdade aqui economiza o seu tempo e o nosso.

TALVEZ AINDA NÃO SEJA A HORA SE

- suas regras quase não mudam e são poucas;
- o volume de decisões é baixo e não há pressão de auditoria;
- a lógica atual, mesmo no código, está estável e ninguém sofre com ela;
- velocidade de produto importa, hoje, mais que governança de decisão.

PROVAVELMENTE É A HORA SE

- mudar uma regra leva semanas e depende da fila do TI;
- você não sabe com segurança qual versão está em produção;
- uma auditoria ou um recurso exige reconstruir uma decisão antiga;
- uma ou duas pessoas concentram a lógica crítica;
- o volume cresce e, com ele, o custo ou o risco;
- você vai espalhar agentes de IA sobre decisões que precisam ser determinísticas.

Se você se viu mais na primeira lista, guarde este guia. Adicionar uma camada de regras agora seria resolver um problema que você ainda não tem. Se você se viu na segunda, a conversa vale a pena.

Como começar sem big bang

A maior objeção contra externalizar regras não é o conceito, é o medo do "big bang": parar tudo e migrar a lógica de uma vez. Esse medo é justificado, e a resposta é não fazer isso.

A forma que recomendamos é incremental. Escolhe-se um processo de decisão real, de preferência um que dói (muda muito, depende do TI, é difícil de auditar). Externaliza-se esse processo primeiro, em paralelo ao que já existe, sem desligar nada. Mede-se o resultado. Só então se decide o próximo.

A PROVA DE CONCEITO QUE PROPOMOS

Duas semanas, com um processo real da sua operação, não um exemplo de demonstração. O objetivo não é impressionar com tela bonita, é responder a uma pergunta concreta: quando a regra sai do código e vai para um motor governado, quanto mais rápido a sua área de negócio consegue mudá-la, e o que isso libera?

Começa pequeno. Cresce pela evidência, não pela fé.

Cinco setores, o mesmo padrão

A tese atravessa setores. Estas são empresas que tiraram a regra de negócio do lugar provisório e deram a ela um endereço próprio. Setores diferentes, o mesmo movimento.

SEGUROS

Sura

Reduziu o tempo de resposta em **40%**, com decisões mais consistentes, alterando parâmetros sem depender da equipe de TI.

FINANCEIRA E VAREJO

Midway

grupo Riachuelo

As áreas passaram a tomar decisões com autonomia e controle, ganhando agilidade sem perder a governança.

LOGÍSTICA

Mundial Logistics

Mudou a dinâmica entre TI e negócio. A operação ficou mais fluida e a área ganhou tempo para o que é estratégico.

LOCAÇÃO

Unidas

De solução de apoio, a gestão de regras virou parte central da operação de precificação.

BENEFÍCIOS

Edenred

Precificação mais ágil e segura no Ticket Combustível, acelerando aprovações e impulsionando as vendas.

Depoimentos públicos divulgados pelas próprias empresas em abaccus.com.br/cases-de-sucesso.

Continue a conversa

Os próximos lugares para se aprofundar, no seu tempo.

ENTENDA A CATEGORIA

O que é um BRMS

abaccus.com.br/abaccus-brms

Site da Abaccus

abaccus.com.br

COMPARE, PLATAFORMA A PLATAFORMA

Abaccus e Drools

abaccus.com.br/alternativa-ao-drools

Abaccus e FICO Blaze Advisor

abaccus.com.br/alternativa-ao-fico-blaze-advisor

Abaccus e IBM ODM

abaccus.com.br/alternativa-ao-ibm-odm

Abaccus e GoRules

abaccus.com.br/alternativa-ao-gorules

Abaccus e Nected


abaccus.com.br/alternativa-ao-nected

IA E DECISÃO DETERMINÍSTICA

Como integrar agentes de IA a uma camada determinística

abaccus.com.br/arquitetura-ia-deterministica

Quando quiser sair do conteúdo e olhar a sua operação, agende um diagnóstico em abaccus.com.br/fale-conosco. Partimos de um processo real seu, não de uma demonstração genérica.



O PRÓXIMO PASSO

Onde mora a sua regra de negócio?

Se depois deste guia a pergunta ficou na sua cabeça, é um bom sinal. O passo natural é olhar para um processo real da sua operação e medir. Em vez de uma demonstração genérica, partimos de uma decisão que importa para você.

Agende um diagnóstico → abaccus.com.br/fale-conosco



Gerencie regras. Reduza custos. Acelere resultados.

BRMS 100% brasileiro · abaccus.com.br · linkedin.com/company/abaccusbrms