

Vantagem	Caracterização
Produtividade	<ul style="list-style-type: none"> • Automatização da geração de código a partir de modelos por meio de uso de ferramentas de transformação, incentivando os desenvolvedores a retornarem às etapas iniciais de requisitos e análise; • Tarefas repetitivas de codificação são implementadas nas transformações, poupando tempo e esforço que podem ser despendidos em outras tarefas.
Portabilidade	<ul style="list-style-type: none"> • A partir de um mesmo modelo pode-se gerar código para diferentes plataformas.
Interoperabilidade	<ul style="list-style-type: none"> • Cada parte do modelo pode ser transformado em código para uma plataforma diferente, o que resulta em um software que pode ser executado em um ambiente heterogêneo, mas que mantém sua funcionalidade global.
Manutenção e documentação	<ul style="list-style-type: none"> • Alterações são realizadas diretamente nos modelos, mantendo-os consistentes como o código-fonte, o qual é gerado automaticamente a partir de transformações aplicadas nesses modelos; • A documentação permanece atualizada, o que facilita as tarefas de manutenção.
Comunicação	<ul style="list-style-type: none"> • Uma vez que os modelos são mais abstratos que o código-fonte, o que não exige conhecimento técnico associado à plataforma de implementação para sua compreensão, os especialistas do domínio podem utilizar diretamente os modelos para identificar mais facilmente as questões associadas ao negócio; • Os especialistas de tecnologia da informação podem identificar os elementos técnicos usando os mesmos modelos.
Reuse	<ul style="list-style-type: none"> • O reuso é feito em nível de modelos ao invés de um nível de código-fonte; • O código-fonte pode ser automaticamente regenerado para novos contextos por meio de ferramentas de transformações apropriadas.
Verificação e Otimizações	<ul style="list-style-type: none"> • Os modelos facilitam a análise por verificadores semânticos, conforme a sintaxe de seu metamodelo, e a execução de otimizações automáticas; • Minimização da ocorrência de erros semânticos, o que fornece implementações mais eficientes.
Correção	<ul style="list-style-type: none"> • Ferramentas de transformação evitam a introdução de erros acidentais, tais como erros de digitação e de sintaxe; • Erros conceituais podem ser identificados em um nível mais alto de abstração.