

Diagrama de componentes UML: o que é, como fazer e exemplos

Por que usar um diagrama UML?

Quero aprender mais sobre diagramas de caso de uso pois são novidade para mim. ↓

Eu quero criar meu próprio diagrama de caso de uso no Lucidchart.

Eu quero criar um diagrama de caso de uso a partir de um modelo do Lucidchart.

Conteúdo

O que é diagrama de componentes UML?

Benefícios do diagrama de componentes

Como usar o diagrama de componentes

Formas e símbolos do diagrama de componentes

Como usar formas e símbolos de componentes

Exemplos de diagrama de componentes

Como criar um diagrama de componentes

O diagrama de componentes é essencial para criar um sistema de software. Eles podem ser feitos usando um software de diagramação UML, e ajudam sua equipe a entender a estrutura de sistemas existentes e a criar sistemas novos. Saiba mais sobre o diagrama de componentes a seguir.

Leitura de 6 minuto(s)

Quer criar um diagrama UML próprio? Experimente o Lucidchart. É rápido, fácil e completamente gratuito.

[Criar um diagrama UML](#)

O que é diagrama de componentes UML?

O diagrama de componentes mostra o relacionamento entre diferentes componentes de um sistema. Para fins de UML 2.0, o termo "componente" refere-se a um módulo de classes que representa sistemas ou subsistemas independentes com capacidade de interagir com o restante do sistema.

Para isso, existe uma abordagem de desenvolvimento em torno de componentes: o desenvolvimento baseado em componentes (CBD). Nela, o diagrama de componentes identifica os diferentes componentes para que todo o sistema funcione corretamente.

O mais comum, em uma abordagem de programação OO, é o desenvolvedor sênior usar o diagrama de componentes para agrupar classes com base em um objetivo comum, para que o desenvolvedor e outras pessoas envolvidas analisem o projeto de desenvolvimento de software de forma generalizada.

Benefícios do diagrama de componentes

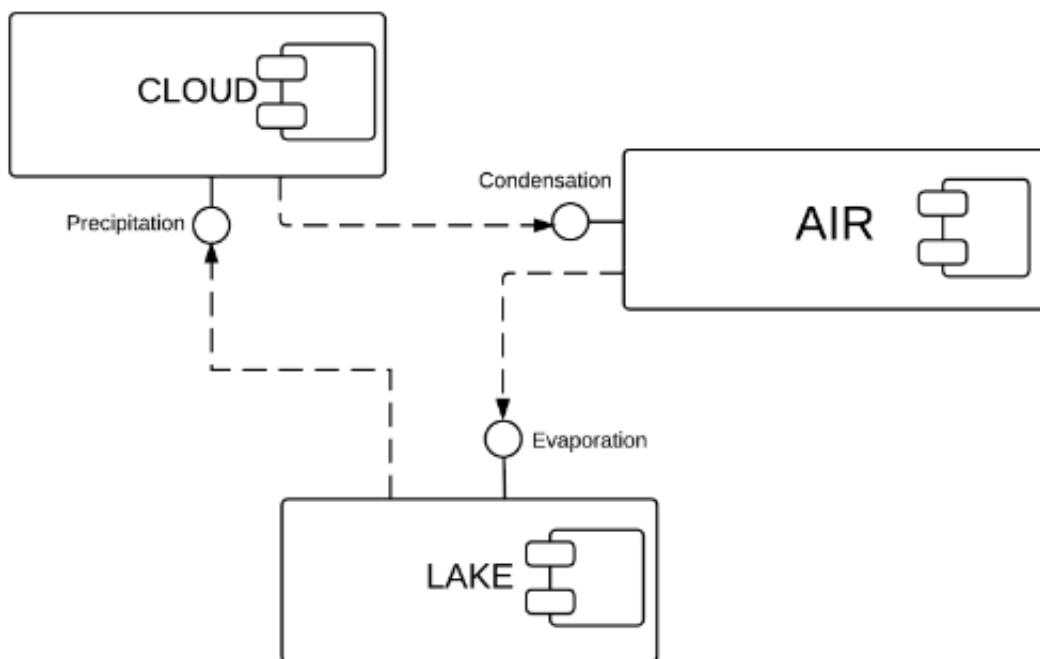
Embora o diagrama de componentes pareça complexo à primeira vista, ele é indispensáveis na criação do seu sistema. Com ele, sua equipe:

- Imagina a estrutura física do sistema.
- Presta atenção aos componentes do sistema e como eles se relacionam.
- Enfatiza o comportamento do serviço quanto à interface.

Como usar o diagrama de componentes

O diagrama de componentes em UML dá uma visão geral do sistema de software. Compreender o comportamento exato do serviço fornecido por cada software fará de você um desenvolvedor melhor. O diagrama de componentes descreve sistemas de software implementados em qualquer linguagem ou estilo de programação.

A Linguagem de modelagem unificada (UML) é um conjunto de convenções para diagramas baseados em objetos e possui várias aplicações. No diagrama de componentes, a UML determina que componentes e pacotes sejam conectados a linhas que representam conectores de montagem e de delegação. Saiba mais sobre UML e seus usos no nosso guia "**O que é UML?**"

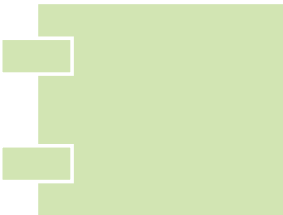

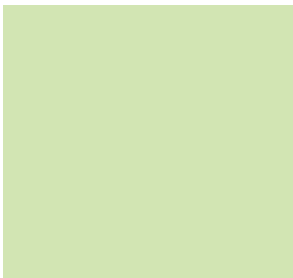


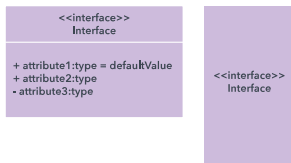
Quer criar um diagrama UML próprio? Experimente o Lucidchart. É rápido, fácil e completamente gratuito.

[Criar um diagrama UML](#)

Formas e símbolos do diagrama de componentes

O diagrama de componentes pode ser simples e geral ou detalhado e complexo. De toda forma, recomendamos conhecer os símbolos UML apropriados. Veja abaixo os tipos de formas mais comuns ao ler e criar um diagrama de componentes:

Símbolo	Nome	Descrição
	Símbolo de componente	Entidade necessária para executar uma função de estereótipo. O componente fornece e consome comportamento em interfaces ou por meio de outros componentes. Considere os componentes como um tipo de classe . Na UML 1.0, o componente tem forma de bloco retangular, com dois retângulos menores saindo pela lateral. Na UML 2.0, o componente tem forma de bloco retangular com uma pequena imagem da forma antiga do diagrama de componentes.
		
	Símbolo de nó	Representa objetos de hardware ou software, de nível superior aos componentes.



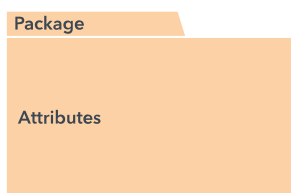
Símbolo de interface

Mostra entradas ou materiais que o componente recebe ou fornece. A interface pode ser representada por notas ou símbolos de texto, como formato de pirulito, soquete e "bola-e-soquete".



Símbolo de porta

Especifica um ponto de interação separado entre o componente e o ambiente. O símbolo da porta é um pequeno quadrado.



Símbolo de pacote

Agrupar vários elementos do sistema e é representado por pastas de arquivos no Lucidchart. Assim como as pastas de arquivos agrupam várias folhas, o pacote pode ser desenhado em torno de vários componentes.



Símbolo de nota

Com ele, os desenvolvedores fixam uma meta-análise no diagrama de componentes.

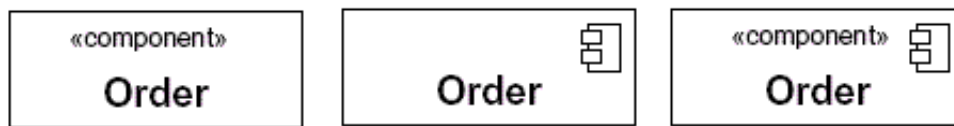


Símbolo de dependência

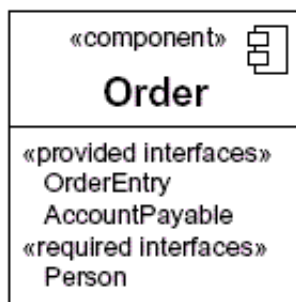
Mostra que uma parte do sistema depende de outra. A dependência é representada por linhas tracejadas que

permissões necessárias que vinculam um componente (ou elemento) a outro.

Como usar formas e símbolos de componentes

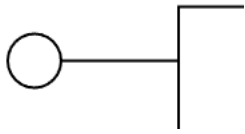


Existem três maneiras comuns de criar o compartimento de nome do componente. É sempre necessário incluir o texto do componente entre aspas angulares e/ou o logotipo do componente. Essa diferença é importante, pois o retângulo com apenas um nome dentro representa classificadores (elementos de classe).



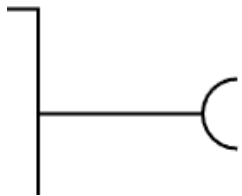
Assim como na notação de classe, o componente também tem um espaço para listar interfaces, semelhante a como atributos e métodos à notação de classe são adicionados. A interface representa os locais onde os grupos de classes no componente se comunicam com outros componentes do sistema. Outra maneira de representar a interface é estender um símbolo da caixa de componentes. Veja um rápido resumo dos símbolos mais usados

componentes. veja um rápido resumo dos símbolos mais usados.



Interface fornecida: uma linha reta a partir da caixa de componentes e com um círculo anexado. Esse símbolo representa a interface na qual um componente produz informações usadas pela interface necessária de outro

componente.



Interface necessária: uma linha reta da caixa de componentes e com um semicírculo anexado (também representado como uma seta tracejada com uma seta aberta). Esse símbolo representa a interface na qual um componente requer informações para executar a função correta.

Na UML, o diagrama de componentes representa visualmente como os componentes de um sistema de software se relacionam. Para criar um, use a biblioteca de formas de diagramas de componentes personalizada do Lucidchart. O diagrama de componentes deve mostrar:

- O escopo do sistema
- A estrutura geral do sistema de software
- As metas que o sistema ajuda entidades humanas ou não humanas (conhecidas como atores) a atingir

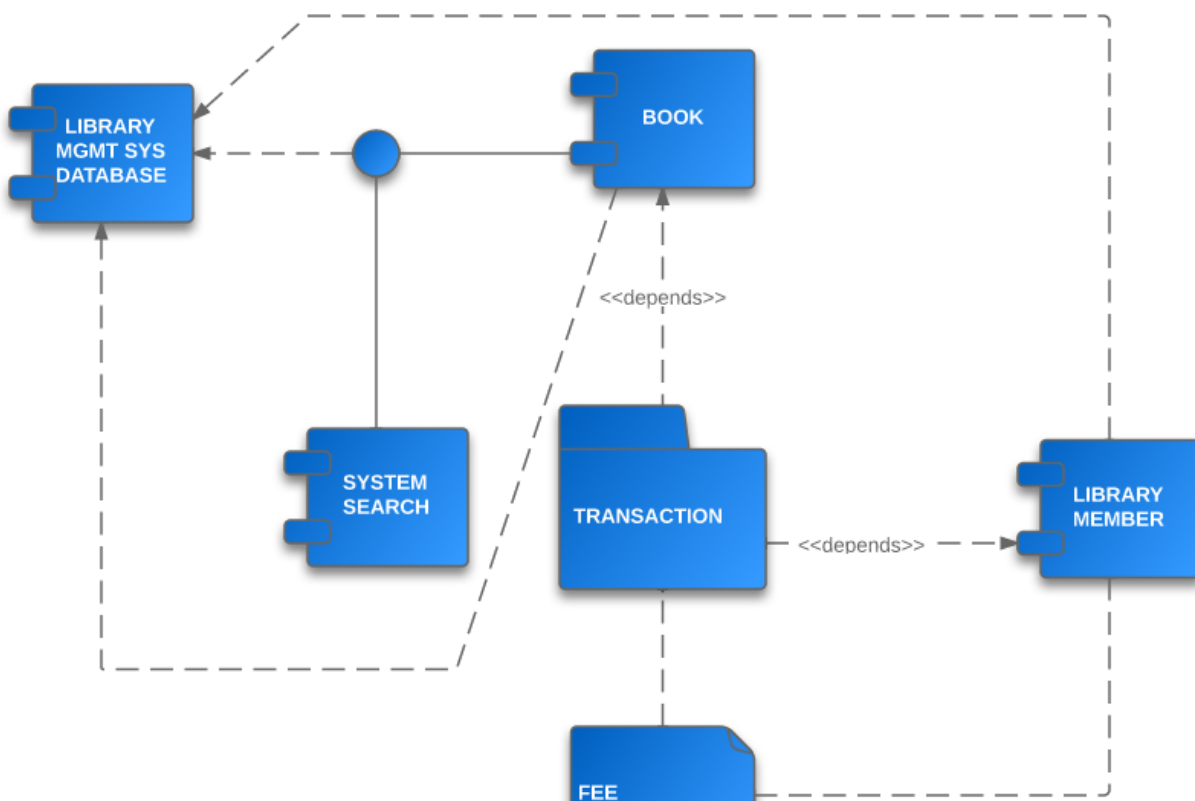
Exemplos de diagrama de componentes

O diagrama de componentes UML simplifica os mais complexos processos. Os exemplos abaixo mostram como usar o diagrama de componentes UML para mapear o comportamento de processos específicos.

Diagrama de componentes para um sistema de gerenciamento de bibliotecas

O sistema de bibliotecas foi um dos primeiros sistemas no mundo a ser amplamente gerenciado por computador. Hoje em dia, muitos desses sistemas são gerenciados na nuvem por serviços terceirizados, não internamente. Embora o termo “sistema de biblioteca” normalmente remeta a uma maneira de monitorar livros impressos, os sistemas de bibliotecas modernos organizam todos os dados registrados por usuários.

Essas transações criam uma rede de relacionamentos entre os componentes do sistema da biblioteca. Veja no diagrama UML abaixo como os relacionamentos e o sistema funcionam. Você ou sua equipe também pode editar este modelo pronto.





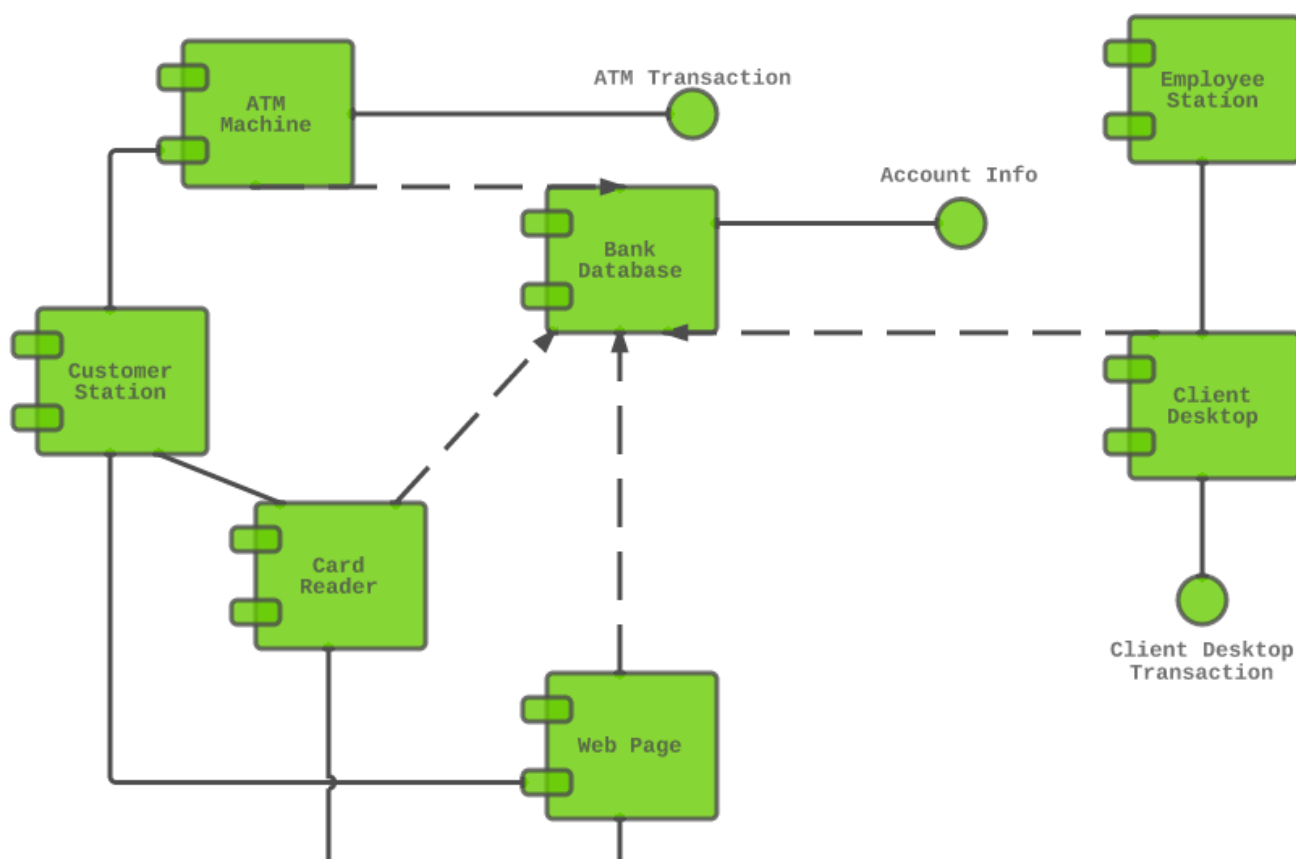
[Clique aqui para usar este modelo](#)

Diagrama de componentes para um sistema de caixa eletrônico

O diagrama de componentes é semelhante ao diagrama de classes, pois mostra como os itens de um determinado sistema se

relacionam. No entanto, o diagrama de componentes representa conexões mais complexas e variadas do que a maioria dos diagramas de classes.

No diagrama abaixo, cada componente é colocado em uma caixa pequena. As linhas pontilhadas com setas mostram como alguns componentes dependem de outros. Por exemplo: o leitor de cartão, a página web, o desktop do cliente e o sistema de caixa eletrônico dependem todos do banco de dados do banco. As linhas pontilhadas com círculos na ponta, conhecidas como símbolos de “pirulito”, indicam uma relação de realização.





[Clique aqui para usar este modelo](#)

Como criar um diagrama de componentes

No Lucidchart, é muito fácil criar um diagrama de componentes UML bastante detalhado do zero. Basta seguir os passos abaixo:

1. Abra um documento em branco ou use um modelo pronto para editar.
2. Ative a biblioteca de formas UML. Clique em "Formas" no lado esquerdo do editor, marque "UML" no Gerenciador de bibliotecas de formas e clique em "Salvar".
3. Selecione a forma desejada na biblioteca que você adicionou e arraste a forma da caixa de ferramentas para a tela de desenho.
4. Desenhe linhas entre as formas para modelar o fluxo de processo.

Veja mais ajuda e um guia detalhado em [Como desenhar um diagrama de componentes em UML](#).

Recursos adicionais

[Tudo sobre diagramas de estrutura de componentes](#)

[Tudo sobre diagramas de estrutura composta →](#)

[O que é um diagrama de sequência UML? →](#)

[O que é um diagrama de máquina de estados? →](#)

[Tudo sobre diagrams de interação UML →](#)

[Tudo sobre diagramas de pacotes UML →](#)

[O que é um diagrama de implementação? →](#)

[Diagrama de caso de uso UML: O que é, como fazer e exemplos →](#)

[O que é um diagrama de objetos? →](#)

[O que é diagrama de atividades UML? →](#)

[O que é um diagrama UML? →](#)

[O que é um diagrama de classe UML? →](#)


[Software de diagramas UML online →](#)

Use o Lucidchart para colaborar e criar diagramas de UML ao criar uma conta gratuita hoje! Sem a necessidade de plugins ou download.

Quer criar um diagrama UML próprio? Experimente o Lucidchart. É rápido, fácil e completamente gratuito.

[Criar um diagrama UML](#)

Iniciar	Produtos	Recursos	Casos de uso	Companhia
Preços	Visão geral do Lucidchart	Campus de Aprendizagem	Educação	Quem somos
Individual	Integrações	Blog	Visualizando organizações e equipes Ágeis	Missão
Equipe	Segurança	Suporte	Criando diagramas técnicos	Liderança
Empresa		Estudos de caso	Documentação de sistemas e arquitetura	Notícias
Falar com área de vendas		Biblioteca de diagramas	Criar mapas de processos e fluxogramas	Carreiras
		Parceiros de soluções		

 Portuguese ▾

[Privacidade](#) [Jurídico](#) [Opções de privacidade de cookies](#)

[Política de cookies](#)

[in](#) [X](#) [@](#) [f](#) [▶](#) [📱](#) [🎵](#)

