

O pacote `fddiagram`*

Rodrigo Smarzaro

<https://github.com/Smarzaro/fddiagram>

2026-09-27

Abstract

O `fddiagram` gera automaticamente, a partir de uma lista textual de Dependências Funcionais no formato `lado_esquerdo -> lado_direito`, o diagrama visual correspondente (uma “régua” horizontal por DF, com marcas nos atributos determinantes e setas nos determinados). O universo de atributos, sua ordem e a posição de cada régua são calculados automaticamente: não é preciso informar coordenadas, larguras ou qual atributo fica mais à esquerda ou à direita de cada DF.

Contents

1	Introdução	1
2	Instalação	2
3	Referência de opções	2
3.1	Estrutura e conteúdo	3
3.2	Geometria	3
3.3	Aparência	4
4	Mensagens de erro	4
5	Limitações conhecidas	4
6	Histórico de versões	5
7	Licença e autoria	5

1 Introdução

Diagramas de Dependências Funcionais (DFs) são úteis para ensinar e verificar normalização de esquemas relacionais, mas desenhá-los à mão em $\text{T}_{\text{E}}\text{X}$ (calculando manualmente qual atributo fica mais à esquerda ou à direita de cada régua, e em que profundidade cada uma deve ficar para não colidir com as outras) é tedioso e propenso a erro. O `fddiagram` resolve isso: você escreve as DFs quase como as escreveria em uma prova ou lista de exercícios, e o pacote calcula o resto.

*Este documento corresponde ao `fddiagram` v0.3, datado de 2026-09-27.

```

fddiagram \begin{fddiagram} [opções]
          <lado esquerdo> -> <lado direito> ;
          <lado esquerdo> -> <lado direito> ;
          :
        \end{fddiagram}

```

O ambiente principal do pacote. Cada linha do corpo descreve uma Dependência Funcional; *<lado esquerdo>* e *<lado direito>* são listas de atributos separadas por vírgula. As DFs são separadas por ponto e vírgula (;); a última pode omitir o ; final. Linhas em branco no meio da lista são toleradas e ignoradas.

Por padrão, o universo de atributos (quais colunas aparecem e em que ordem) é derivado automaticamente: é a união de todos os atributos citados, na ordem da primeira aparição. A chave `attrs` (seção 3) permite fixar essa lista explicitamente.

Cada DF ganha sua própria régua horizontal, na ordem em que foi escrita, rotulada `df1`, `df2`, ... automaticamente.

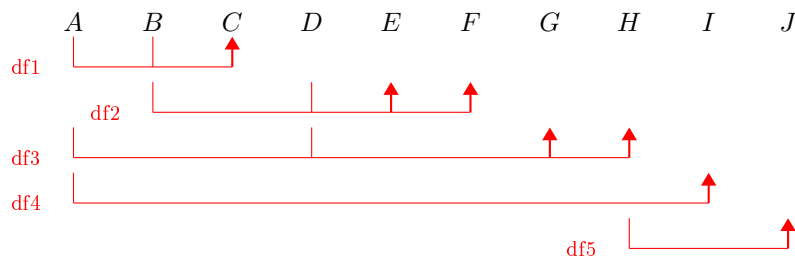
Exemplo mínimo:

```

\begin{fddiagram}
  A,B -> C ;
  B,D -> E,F ;
  A,D -> G,H ;
  A -> I ;
  H -> J
\end{fddiagram}

```

que produz:



2 Instalação

Coloque `fddiagram.sty` em algum diretório que o $\text{\TeX}$ consulte (por exemplo, na mesma pasta do seu documento, ou em uma árvore pessoal do $\text{\TeX}$ MF) e carregue normalmente:

```
\usepackage{fddiagram}
```

O pacote depende de `tikz` (com a biblioteca `arrows.meta`), `xparse` e `l3keys2e`, todos presentes em qualquer distribuição $\text{\TeX}$ Live ou $\text{\MiKTeX}$ atual, carregados automaticamente pelo `fddiagram`.

3 Referência de opções

Todas as chaves abaixo podem ser dadas tanto no carregamento do pacote,

`\usepackage[chave=valor,...]{fddiagram}`

fixando um novo padrão para *todos* os diagramas do documento, quanto em cada ambiente,

`\begin{fddiagram}[chave=valor,...]`

valendo só para aquele diagrama e sobrescrevendo, só ali, o padrão dado no `\usepackage`.

3.1 Estrutura e conteúdo

attrs `attrs = {\i>lista separada por vírgulas}`

Fixa o universo de atributos e sua ordem explicitamente, em vez de derivar automaticamente a partir do texto. Útil para incluir um atributo que não participa de nenhuma DF, ou para forçar uma ordem diferente da ordem de aparição. Padrão: vazio (deriva automaticamente).

```
\begin{fddiagram}[attrs={A,B,C,D,E}]
  A,B -> C ;
  C,D -> E ;
  D,E -> B
\end{fddiagram}
```

labelalign `labelalign = local | left`

Controla o alinhamento horizontal dos rótulos (`df1`, `df2`, ...). Com `local` (padrão), cada rótulo fica colado ao início da própria régua. Com `left`, todos os rótulos ficam alinhados em uma única coluna, na margem esquerda de todo o diagrama.

3.2 Geometria

spacing Distância horizontal entre atributos consecutivos, em cm. Aumente para nomes de atributo longos. Padrão: 1.05.

rowheight Distância vertical entre as réguas de DFs consecutivas, em cm. Padrão: 0.6.

stublen Comprimento das marcas curtas (traços nos atributos determinantes, setas nos determinados) coladas à régua, em cm. Padrão: 0.4.

3.3 Aparência

color Cor das réguas, marcas, setas e rótulos. Aceita qualquer cor conhecida pelo `xcolor/tikz`. Padrão: `red`.

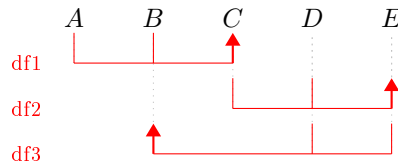
arrow Estilo da ponta de seta (biblioteca `arrows.meta` do TikZ) usado nos atributos determinados. Padrão: `Triangle`. Outras opções úteis: `Stealth`, `{Triangle[open]}`, `Latex`.

guides Booleano. Se `true`, liga cada marca (traço ou seta) até o nome do atributo com uma linha-guia fina, terminando exatamente na borda inferior do rótulo (sem sobrepor o texto). Padrão: `false`.

As linhas-guia de *todas* as DFs são desenhadas antes de qualquer régua, marca ou seta: onde um cruzamento com a régua de outra DF for inevitável, a régua (desenhada por cima, na camada seguinte) aparenta continuidade, sem interrupção visual.

guidecolor Cor da linha-guia (usada apenas quando `guides=true`). Padrão: `lightgray`.

guidestyle Estilo de tracejado da linha-guia: `dashed` ou `dotted`. Padrão: `dotted`.



4 Mensagens de erro

Se uma linha do corpo do ambiente não contiver exatamente um `->`, o `fddiagram` emite um erro apontando a linha problemática e segue desenhando o restante do diagrama normalmente, em vez de interromper a compilação com um erro críptico do `TEX`:

```
! Package fddiagram Error: A dependencia funcional '...' nao
(fddiagram)                contem exatamente um '->' (confira se nao
(fddiagram)                falta ou sobra um '->' nessa linha).
```

5 Limitações conhecidas

- Nomes de atributo não podem conter as substrings `->`, `,` ou `;`, pois são os delimitadores da sintaxe.
- Cada DF ocupa sua própria linha, na ordem em que foi escrita; não há compactação automática de réguas cujos intervalos não se sobrepõem.

6 Histórico de versões

- v0.1** Primeira versão: ambiente `fddiagram`, derivação automática do universo de atributos, opções `attrs`, `spacing`, `rowheight`, `stublen`, `color`, `arrow`.
- v0.2** Opções também no carregamento do pacote (via `l3keys2e`); correção de linha em branco no meio da lista de DFs; `stublen` padrão alterado para 0.4; novas opções `labelalign`, `guides`, `guidecolor`, `guidestyle`; mensagens de erro amigáveis para DF malformada.
- v0.3** Linhas-guia agora terminam exatamente na borda inferior do rótulo do atributo (âncora `.south`), sem sobrepor o texto; correção de ordem de desenho (*z-order*): todas as linhas-guia passaram a ser desenhadas antes de qualquer régua/marca/seta, numa camada própria, para que cruzamentos inevitáveis não interrompam visualmente a régua; cabeçalho de licença e autoria.

7 Licença e autoria

Copyright © 2026 Rodrigo Smarzaro. Distribuído sob a [LaTeX Project Public License](#), versão 1.3c ou posterior; veja o arquivo LICENSE que acompanha o pacote.

Repositório, contato e relato de problemas:

<https://github.com/Smarzaro/fddiagram>