

Funktionen und Graphen 2D, Level 2

Prof. Dr. Dörte Haftendorn, MuPAD 4, <http://haftendorn.uni-lueneburg.de> Aug.06

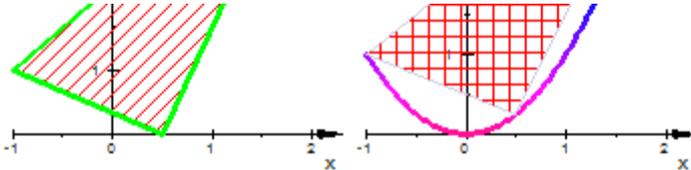
Automatische Übersetzung aus MuPAD 3.11, (Vorversion 6.12.01) Version vom 9.09.05

Es fehlen noch textliche Änderungen, die MuPAD 4 direkt berücksichtigen, das ist in Arbeit.

Web: <http://haftendorn.uni-lueneburg.de>

www.mathematik-verstehen.de

+++++



Die Kenntnis von Graphen2d Funktionen, Grundlage, Level 1 wird vorausgesetzt.

LLLLLLLLLLLL-----Level 2-----LLLLLLLLLLLL

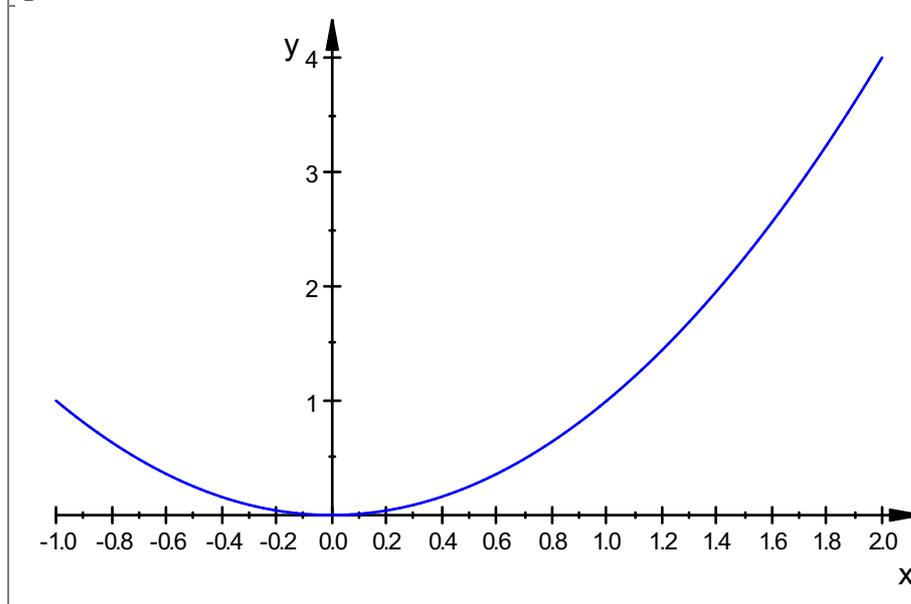
Um Graphen verschiedenen Typs kombinieren zu können, gibt es in MuPAD die Möglichkeit

sogenannte "Graphik-Primitive" zu erzeugen. Das sind Graphik-Bausteine, die selbst nicht sichtbar sind,

sondern erst durch die Funktion plot dargestellt werden.

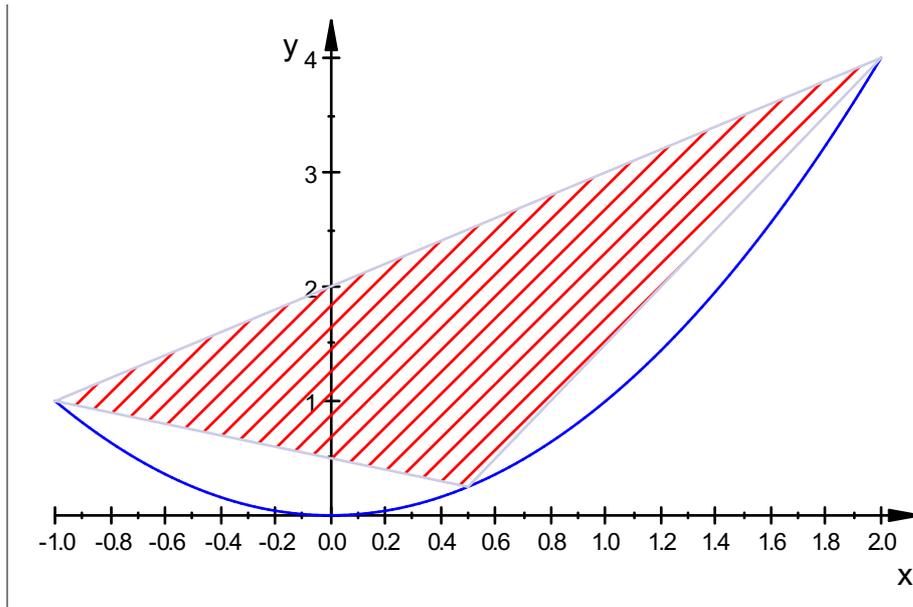
Einfachste Version ist:

```
fkt:=plot::Function2d(x^2,x=-1..2):  
plot(fkt)
```



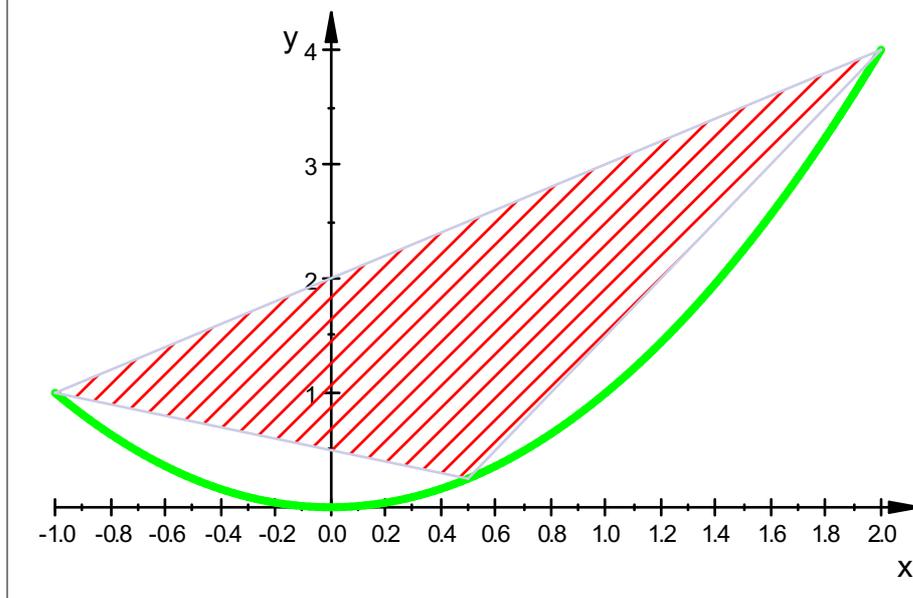
Das lohnt sich so noch nicht, aber wenn man nun darein etwas zeichnen will, geht das so:

```
drei:=plot::Polygon2d([[ -1,1],[0.5,0.25],[2,4],[ -1,1]],  
Filled=TRUE, Color=[0.8,0.8,0.9]):  
plot(fkt,drei)
```



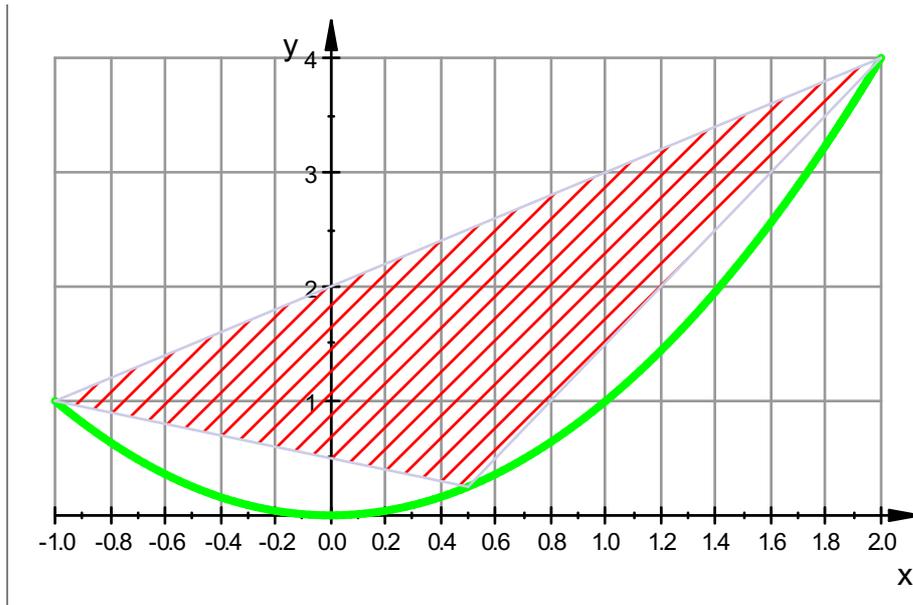
Sowohl bei `plotfunc2d(%?)` als auch hier kann man Farben ändern kann.

```
fkt:=plot::Function2d(x^2,x=-1..2,LineWidth=1, Color=RGB::Green):
plot(fkt,drei)
```



Optionen für die Gesamtansicht, die Szene, kann man folgendermaßen beeinflussen:

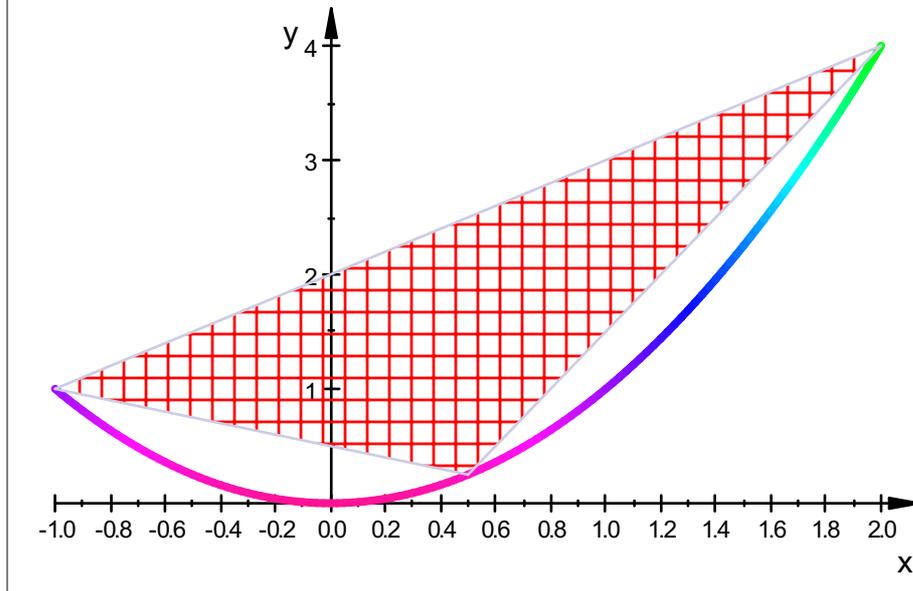
```
sc:=plot::Scene2d(fkt, drei,GridVisible=TRUE):
plot(sc)
```



Grundsätzlich kann man Eigenschaften der Szene ändern.

Auch die Eigenschaften der Graphikprimitiven kann man ändern:

```
fktneu:=plot::modify(fkt,LineColorType=Rainbow):
dreineu:=plot::modify(drei,FillPattern=CrossedLines):
plot(fktneu,dreineu)
```



Die Ideen, wie die Optionen geschrieben werden müssen, kann man sich aus dem interaktiven

Menu holen. (Doppelklick auf Graphik, dann rechts)

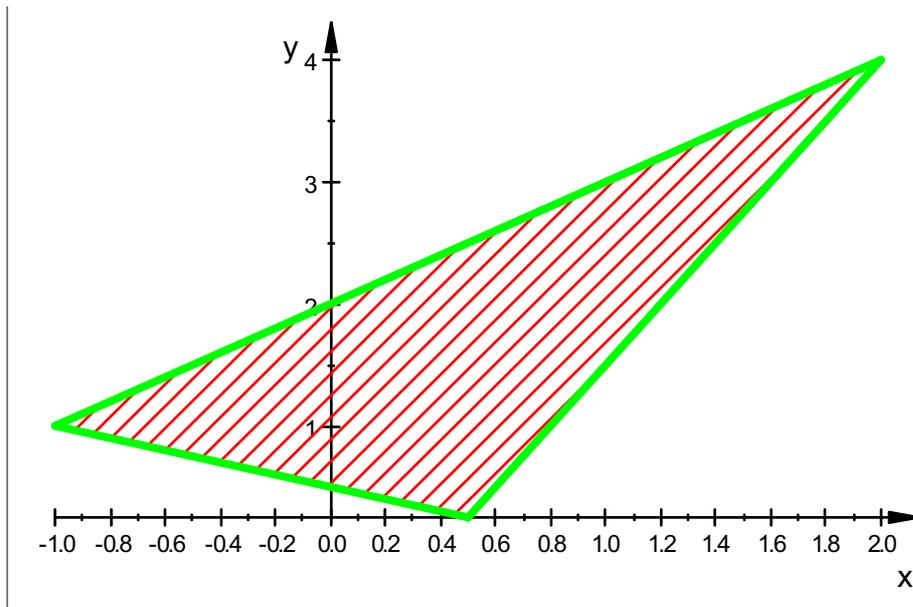
Die Graphikprimitiven sind Objekte, an deren Eigenschaften man auch ähnlich wie in Programmiersprachen herankommt:

```
drei::Color
```

```
[0.8, 0.8, 0.9]
```

3

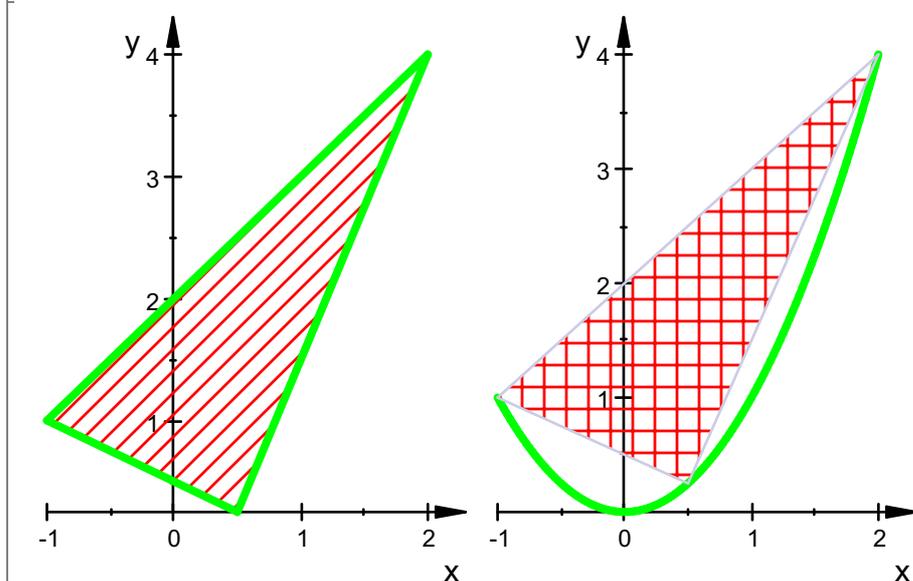
```
drei::Color:=[0,1,0]:drei::LineWidth:=1:plot(drei) //neu definieren
```



Die "Szene" wird zu jeder Graphik erzeugt, Graphikprimitive, die in einem gemeinsamen Koordinatensystem erscheinen sollen, werden in einer Szene zusammengefasst.

Definiert man aber ausdrücklich Szenen für die Einzelgraphen, so stellt plot die Szenen nebeneinander dar:

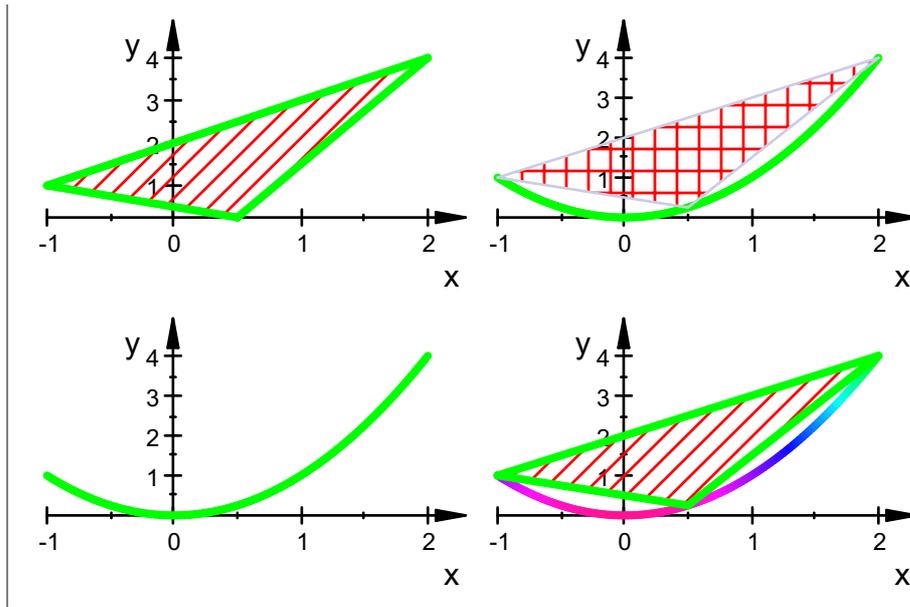
```
sc1:=plot::Scene2d(drei) : sc2:=plot::Scene2d(fkt,dreineu) : plot(sc1,
```



```
sc3:=plot::Scene2d(fkt) : sc4:=plot::Scene2d(fktneu,drei) :
```

Für viele Szenen wird ein passende "Layout" gewählt, das man aber auch beeinflussen kann, wenn man das möchte.

```
plot(sc1, sc2, sc3, sc4)
```



Die Szenen müssen dieselbe Dimension haben.