Template functions

Function f in `d(f,x)` does not have to be defined, it can be a template function with just a name and an argument list. The argument list determines the result. For example, `d(f(x),x)` evaluates to itself because f depends on x. However, `d(f(x),y)` evaluates to zero because f does not depend on y.

Example 1. $f(x)$ depends on x.

```
d(f(x),x)
```

$\mathrm{d}(f(x), x)$

Example 2. $f(x)$ does not depend on y.

```
d(f(x),y)
```

0

Example 3. $f(x, y)$ depends on both x and y.

```
d(f(x,y),y)
```

$\mathrm{d}(f(x, y), y)$

Example 4. $f()$ is shorthand for dependence on any symbol.

```
d(f(),t)
```

$\mathrm{d}(f(), t)$

As the final example shows, an empty argument list causes `d` to evaluate to itself, regardless of the second argument.

Template functions are useful for experimenting with differential forms. For example, verify the identity

$$\nabla \cdot (\nabla \times \mathbf{F}) = 0$$

```
F = (Fx(),Fy(),Fz())
div(curl(F))
```

0