DIGITAL
TECHNOLOGY
GROUP

CamFort

CamFort is our refactoring tool for Fortran providing:

- data gathering on the programming patterns common in scientific models;
- automatic checking & refactoring to improve the code quality of existing models.

Units-of-measure

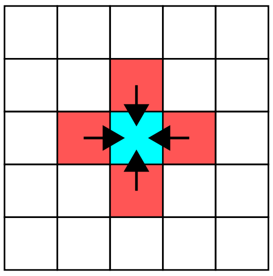
Code that mixes up physical units can produce bogus output, leading to spectacular failures such as the Mars Climate Orbiter mission, or forced retractions of submitted scientific papers.

```
!= unit(N s) :: p1
!= unit(lbf s) :: p2
real :: p1, p2

! ...units-checker will detect an error here:
p1 = p2
```

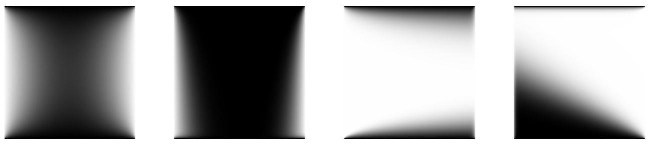
CamFort sets up and solves a set of constraints between unit specifications, so it understands combinations of units in formulas and can infer the units in many cases without explicit annotation.

```
!= unit(m) :: x
!= unit(s) :: t
!= unit(m/s) :: v1
real :: x, t, v1, v2

! v2 inferred with unit (m/s)
v2 = x / t
v1 = v2
```

Stencil specifications



A common pattern in scientific code is the "stencil" where each cell in an array is updated with an equation using the adjacent cells. These stencils can involve writing very error-prone indexing code. Even just an off-by-1 mistake can produce radically different output, as shown:

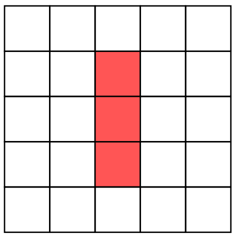


Sample 2-D output of very similar stencil codes

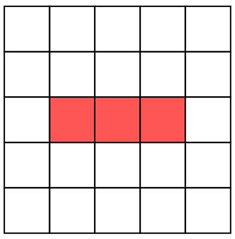
A stencil annotation like this might appear in code linking a variable with a region:

```
!= centered(depth=1, dim=1) :: a
a(i,j) = (a(i-1,j) + a(i,j) + a(i+1,j)) / 3
```

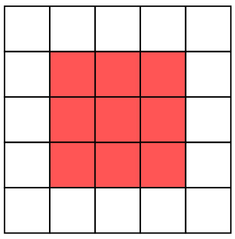
Regions can be combined to form more complex descriptions:



centered(depth=1, dim=1)

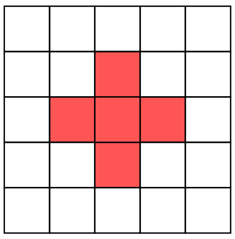


centered(depth=1, dim=2)



centered(depth=1, dim=1)
*
centered(depth=1, dim=2)

Conjunction



centered(depth=1, dim=1)
+
centered(depth=1, dim=2)

Disjunction

For more information:
www.cl.cam.ac.uk/research/dtg/camfort
Source code:
www.github.com/camfort

UNIVERSITY OF CAMBRIDGE