

Taskyield Directive

RZ: Christian Terboven Folie 18

The taskyield Directive



- ▶ The taskyield directive specifies that the current task can be suspended in favor of execution of a different task.
 - ▶ Hint to the runtime for optimization and/or deadlock prevention

C/C++	Fortran
<pre>#pragma omp taskyield</pre>	!\$omp taskyield

Example



taskyield

RZ: Christian Terboven Folie 20

taskyield Example (1/2)



```
#include <omp.h>
void something useful();
void something critical();
void foo(omp_lock_t * lock, int n)
{
   for(int i = 0; i < n; i++)
      #pragma omp task
         something useful();
         while( !omp_test_lock(lock) ) {
            #pragma omp taskyield
         something critical();
         omp_unset_lock(lock);
      }
```

taskyield Example (2/2)



```
#include <omp.h>
void something useful();
void something critical();
void foo(omp lock t * lock, int n)
{
   for(int i = 0; i < n; i++)
      #pragma omp task
          something useful();
                                                      The waiting task may be
          while( !omp test lock(lock) ) {
                                                   suspended here and allow the
             #pragma omp taskyield <</pre>
                                                    executing thread to perform
                                                     other work. This may also
                                                     avoid deadlock situations.
          something_critical();
          omp unset lock(lock);
```

RZ: Christian Terboven