
Subject: Re: RE: Job package: A scope-bound worker thread for non-blocking operations.

Posted by [mirek](#) on Tue, 10 Oct 2017 18:51:14 GMT

[View Forum Message](#) <> [Reply to Message](#)

There is a tricky catch with IsFinished:

```
template <class Range>
ValueTypeOf<Range> ASum(const Range& rng, const ValueTypeOf<Range>& zero)
{
    int n = rng.GetCount();
    if(n == 1)
        return rng[0];
    if(n == 0)
        return 0;
    auto l = Async([&] { return ASum(SubRange(rng, 0, n / 2)); });
    auto r = Async([&] { return ASum(SubRange(rng, n / 2, n - n / 2)); });
    while(!l.IsFinished() || !r.IsFinished())
        Sleep(1);
    return l.Get() + r.Get();
}
```

What do you think is wrong with this code?

Mirek
