Subject: Interesting advantage of using indices instead of iterators
Posted by mirek on Wed, 10 Jan 2007 10:52:06 GMT
View Forum Message <> Reply to Message

From alt.comp.lang.c++.moderated:

Jeff Koftinoff wrote:

> My only issue with STL's iterator design is that the pointer-style
> based algorithms make it a lot tougher for the compiler to optimize the
> resulting code, taking advantage of cpu pipelining and parallel
> execution units - especially without having the 'restrict' keyword to
> tell the compiler that the dereferenced iterators are not aliasing
> other variables used within the loop. ie, with array indexing the
> compiler can easily tell that certain lines of code can be
> parallelized, but changing a pointer in the loop makes it very hard on
> the compiler designers to track what optimizations can happen. In my
> code on one platform, I've found I could optimize the loop time from
> 170 clock cycles per iteration down to 10 just by using array indexing
> instead of pointer increments! (TMSC6701 DSP, VLIW)

I had many reasons to avoid iterators, but this one is new to me )

--
Mirek Fidler
U++ team leader. http://www.ultimatepp.org

Subject: Re: Interesting advantage of using indices instead of iterators
Posted by Balage on Wed, 10 Jan 2007 11:24:26 GMT
View Forum Message <> Reply to Message

I've also read a lot about this. It's not just iterators.

Many newcomers (including myself at that time) try to use pointer arithmetic to save that addition on array indexing inside loops. But they all fail to realize, that the compiler can do way much better optimizations if it knows what is just read from memory, and what is updated (and also when). Pointer arithmetic makes it a lot harder to figure that out, as the compiler cannot easily track where a pointer currently points to.
With regular array indexing, it exactly knows what element, when, and how is being accessed, and that's very good for optimization.

Subject: Re: Interesting advantage of using indices instead of iterators
Posted by unodgs on Wed, 10 Jan 2007 11:47:38 GMT

Perhaps there are more important reasons (like that one you mentioned) but I don't like iterators because of long declarations...

map<string, const Point&>::const_iterator
vector<int>::iterator

God, what's that?
That's one of reasons why auto is comming up. Indexes are much more natural IMO.