Subject: A new container in works: Flex - fast insertion vector
Posted by mirek on Wed, 29 Aug 2007 17:00:33 GMT
View Forum Message <> Reply to Message

Recently I have enjoyed some time implementing my former idea of novelty random access container (aka indexed) container, similar to Vector or std::vector, which has much improved time of insertion of element at arbitrary position, while keeping operator[] complexity at O(1).

It was a success, my idea seems to work as expected:

I have benchmarked it by creating a Vector/std::vector/Flex by inserting "size" elements at random positions:

int

size (times)	std::vector	Vector	Flex
10 (100000x)	0.078 s	0.016 s	0.031 s
20 (50000x)	0.078 s	0.032 s	0.047 s
40 (25000x)	0.062 s	0.047 s	0.047 s
80 (12500x)	0.078 s	0.047 s	0.062 s
160 (6250x)	0.094 s	0.047 s	0.078 s
320 (3125x)	0.109 s	0.063 s	0.094 s
640 (1562x)	0.140 s	0.110 s	0.125 s
1280 (781x)	0.203 s	0.156 s	0.172 s
2560 (390x)	0.312 s	0.282 s	0.203 s
5120 (195x)	0.562 s	0.516 s	0.234 s
10240 (97x)	1.078 s	1.032 s	0.250 s
20480 (48x)	2.547 s	2.500 s	0.312 s
40960 (24x)	5.625 s	5.594 s	0.437 s
81920 (12x)	11.672 s	11.625 s	0.578 s
163840 (6x)	23.703 s	23.657 s	0.703 s
327680 (3x)	47.688 s	47.656 s	0.953 s

String

size (times)	std::vector	Vector	Flex
10 (100000x)	0.125 s	0.063 s	0.094 s
20 (50000x)	0.140 s	0.078 s	0.078 s
40 (25000x)	0.141 s	0.063 s	0.078 s
80 (12500x)	0.157 s	0.093 s	0.110 s
160 (6250x)	0.218 s	0.125 s	0.125 s
320 (3125x)	0.329 s	0.187 s	0.188 s
640 (1562x)	0.562 s	0.313 s	0.265 s
1280 (781x)	1.063 s	0.593 s	0.297 s
2560 (390x)	2.047 s	1.203 s	0.344 s
5120 (195x)	3.938 s	2.796 s	0.422 s
10240 (97x)	7.797 s	6.235 s	0.500 s

20480 (48x)	15.140 s	12.641 s	0.656 s
40960 (24x)	30.234 s	25.688 s	0.969 s
81920 (12x)	60.703 s	51.765 s	1.313 s
163840 (6x)	149.062 s	135.922 s	2.078 s
327680 (3x)	433.078 s	411.735 s	3.734 s

Means it is as almost as fast as Vector to about 1000 elements, then it starts to outperform it more and more...

There is a lot of work left before this can be introduced to the Core (it is hard as hell to implement, my productivity doing this is about 10 lines / day, so far I have only implemented Insert, Remove will cost me another day or two... , but nevertheless I am pleased...

Mirek