

Allocators

TODO

Henruch Lauko, Jíří Novotný, Katarína Kejstová

May 18, 2016

Motivation



C++ Allocators API

- ▶ allocated memory representation:

```
struct Block { void* ptr; size_t size; }
```

- ▶ requirements on allocator:

```
Block allocate(size_t)
```

```
void deallocate(Block)
```

```
bool owns(Block)
```

Allocators

- ▶ NullAllocator
- ▶ Mallocator
- ▶ StackAllocator
- ▶ FallbackAllocator
- ▶ Freelist
- ▶ Segregator
- ▶ AffixAllocator
- ▶ StatisticCollector
- ▶ BitmappedBlock

- ▶ alignment?

FallbackAllocator

```
template <typename Primary, typename Fallback>
struct FallbackAllocator {
    Block allocate(size_t size) {
        Block ptr = _primary.allocate(size);
        return ptr ? ptr : _fallback.allocate(size);
    }
    ...
private:
    Primary _primary;
    Fallback _fallback;
}
```

FallbackAllocator - owns

```
template <typename P, typename F>
bool FallbackAllocator<P,F>::owns(const Block& blk) {
    return _primary.owns(blk) || _fallback.owns(blk);
}
```

- ▶ relies on MDFINAE – method definition failure is not an error

Freelist

```
template <class A, size_t min, size_t max, size_t capacity>
struct Freelist {
    Block allocate(size_t size) {
        if (is_inside_bounds(size)) {
            if (_root)
                return pop();
            else
                return _parent.allocate(max);
        }
        return _parent.allocate(size);
    }
    ...
private:
    A _parent;
    struct Node{ Node* next} _root;
}
```


- ▶ TODO problems

Segregator

```
template <size_t threshold,  
          typename SmallAllocator,  
          typename LargeAllocator>  
struct Segregator;
```

- ▶ sizes \leq threshold are managed by SmallAllocator
- ▶ others by LargeAllocator

Composability

```
using Allocator =  
    Segregator<1024,  
        Segregator<512,  
            Freelist<Mallocator, 0, 512,  
                Freelist<Mallocator, 513, 1024>  
        >,  
        Mallocator  
    >;
```


BitmappedBlock

Modularity - composability

- ▶ composition of allocators, specialized by block sizes
- ▶ arrays, lists, trees of allocators

Example

Benchmarks