

# Memory pool

---

**Memory pools**, also called fixed-size blocks allocation, is the use of pools for memory management that allows dynamic memory allocation comparable to malloc or C++'s operator new. As those implementations suffer from fragmentation because of variable block sizes, it is not recommendable to use them in a real time system due to performance. A more efficient solution is preallocating a number of memory blocks with the same size called the **memory pool**. The application can allocate, access, and free blocks represented by handles at run time.

Many real-time operating systems use memory pools, such as the Transaction Processing Facility.

Some systems, like the web server Nginx, use the term *memory pool* to refer to a group of variable-size allocations which can be later deallocated all at once. This is also known as a *region*; see region-based memory management.

## Contents

---

[\*\*Simple memory pool implementation\*\*](#)

[\*\*Memory pool vs malloc\*\*](#)

[\*\*See also\*\*](#)

[\*\*External links\*\*](#)

## Simple memory pool implementation

---

A simple memory pool module can allocate, for example, three pools at compile time with block sizes optimized for the application deploying the module. The application can allocate, access and free memory through the following interface:

- Allocate memory from the pools. The function will determine the pool where the required block fits in. If all blocks of that pool are already reserved, the function tries to find one in the next bigger pool(s). An allocated memory block is represented with a handle.
- Get an access pointer to the allocated memory.
- Free the formerly allocated memory block.
- The handle can for example be implemented with an unsigned `int`. The module can interpret the handle internally by dividing it into pool index, memory block index and a version. The pool and memory block index allow fast access to the corresponding block with the handle, while the version, which is incremented at each new allocation, allows detection of handles whose memory block is already freed (caused by handles retained too long).

## Memory pool vs malloc

---

### Benefits

- Memory pools allow memory allocation with constant execution time. The memory release for thousands of objects in a pool is just one operation, not one by one if *malloc* is used to allocate

memory for each object.

- Memory pools can be grouped in hierarchical tree structures, which is suitable for special programming structures like loops and recursions.
- Fixed-size block memory pools do not need to store allocation metadata for each allocation, describing characteristics like the size of the allocated block. Particularly for small allocations, this provides substantial space savings.
- Allows deterministic behavior on real-time systems avoiding the out of memory errors.

## Drawbacks

- Memory pools may need to be tuned for the application which deploys them.

## See also

---

- Free list
- Object pool
- Slab allocation

## External links

---

- Fast Efficient Fixed-Sized Memory Pool ([http://www.thinkmind.org/download.php?articleid=computation\\_tools\\_2012\\_1\\_10\\_80006](http://www.thinkmind.org/download.php?articleid=computation_tools_2012_1_10_80006))
  - PJLIB Reference: Fast Memory Pool ([http://www.pjsip.org/pjlib/docs/html/group\\_\\_PJ\\_\\_POOL\\_\\_GROUP.htm](http://www.pjsip.org/pjlib/docs/html/group__PJ__POOL__GROUP.htm))
  - A Memory Allocator (<http://gee.cs.oswego.edu/dl/html/malloc.html>)
  - Programming with Memory Pools (<http://svnbook.red-bean.com/en/1.1/ch08s05.html>)
- 

Retrieved from "[https://en.wikipedia.org/w/index.php?title=Memory\\_pool&oldid=768252088](https://en.wikipedia.org/w/index.php?title=Memory_pool&oldid=768252088)"

---

This page was last edited on 2 March 2017, at 18:33 (UTC).

Text is available under the Creative Commons Attribution-ShareAlike License; additional terms may apply. By using this site, you agree to the Terms of Use and Privacy Policy. Wikipedia® is a registered trademark of the Wikimedia Foundation, Inc., a non-profit organization.