



Internship Memory Management

D1 Meeting, 12.11.2020

D1: Prerequisites

Project Setup

General setup:

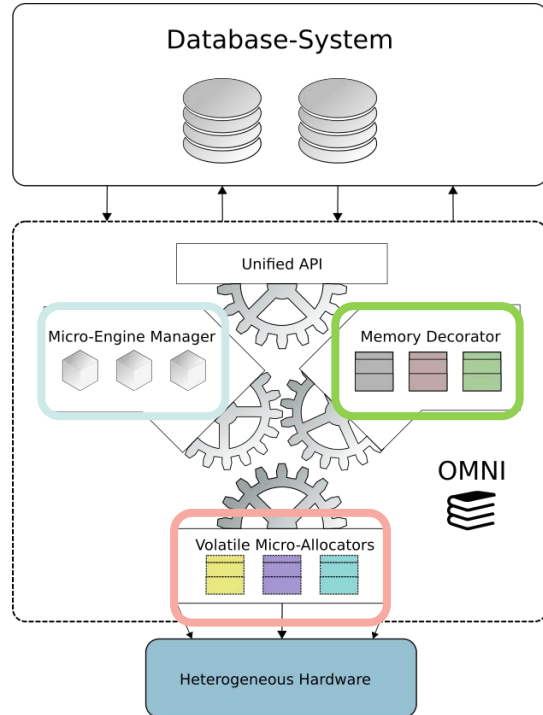
- c++17
- Minimum cmake version 3.18
- ctest
- Internal Git project with branch for every group and deliverable
- Licence: Gnu GPL v3

Project requirements:

- automated detection of hardware features
- stand-alone libraries for every hardware
- Integration of hardware specific libraries

General Architecture

Concept



ME-Managers

- Actual memory management components
- Stateful entities which live as long as the system lives
- Possible Manager could be:
 - Arena Manager (for columns)
 - Slab Manager (for pointers to columns,...)

Allocators

- actually obtain / release memory
- no management code
- static functions (stateless)
- use dedicated specs to address special needs of specific hardware

Decorators

- Functions (or functors) which are used by the micro-engine managers to modify a specific memory region
- Possible use-cases could be:
 - Alignment of memory
 - Debug information like LOC of allocation
 - Fences
 - ...

General procedure

Deliverables

D1	Prerequisites
	<ul style="list-style-type: none">• Project Setup• Build system setup
D2	Allocators Design/Implementation (D/I)
	<ul style="list-style-type: none">• Design of unified allocator interface• Implementation of group-specific allocator
D3	Decorators D/I
	<ul style="list-style-type: none">• Design of unified decorator interface• Implementation of group-specific decorators
D4	Micro-Engine Memory Manager D/I
	<ul style="list-style-type: none">• Design of unified memory manager interface• Implementation of group-specific manager
D5	Wrap Up and Presentation

Schedule

- Every deliverable (except for D5) deadline includes a **short** group-wise presentation followed by discussions
- After the discussion, the next D* is introduced by the tutors
- One week after the start of every D*, a joint discussion is scheduled

Deliverable	Deadline
D1	10.11.2020
D2	26.11.2020
D3	
D4	
D5	



D2: Allocators

D2: Allocators Design/Implementation (D/I)

What do we expect?

