

Slab Allocation

Dr Andrew Scott

a.scott@lancaster.ac.uk

1

Slab Allocation

- Allocate ‘large’ memory area, e.g. using Buddy
- Sub-divide area into common sizes as *slabs*
 - Objects allocated to best slab based on
 - Object size
 - Memory word or cache alignment
 - Many processors require base types to be word aligned
 - Don’t want object split across cache lines (if we can avoid it)
 - Any inter-object (guard) spacing to catch write overruns

2

Slab Allocation

- More generally, allocate common memory sizes
 - Similar sized (not exactly same type) objects can share slab

Buddy Allocator gets pages from Block Allocator

The diagram illustrates the slab allocation process. It starts with a 'Buddy' block, which is linked to a 'Buddy' block that receives input from a 'Buddy Allocator' (noted as getting pages from a 'Block Allocator'). This 'Buddy' block then branches into three categories of slabs: 'Full Slabs', 'Partially Full Slabs', and 'Empty Slabs', all labeled as 'Slabs of size j Kbytes'. The 'Partially Full Slabs' are further detailed with a 'head' and a 'tail'. The 'Full Slabs' and 'Partially Full Slabs' are shown containing multiple 'Object' blocks. A 'Key benefits' section lists 'Little fragmentation' and 'Fast allocation for objects'.

3
