

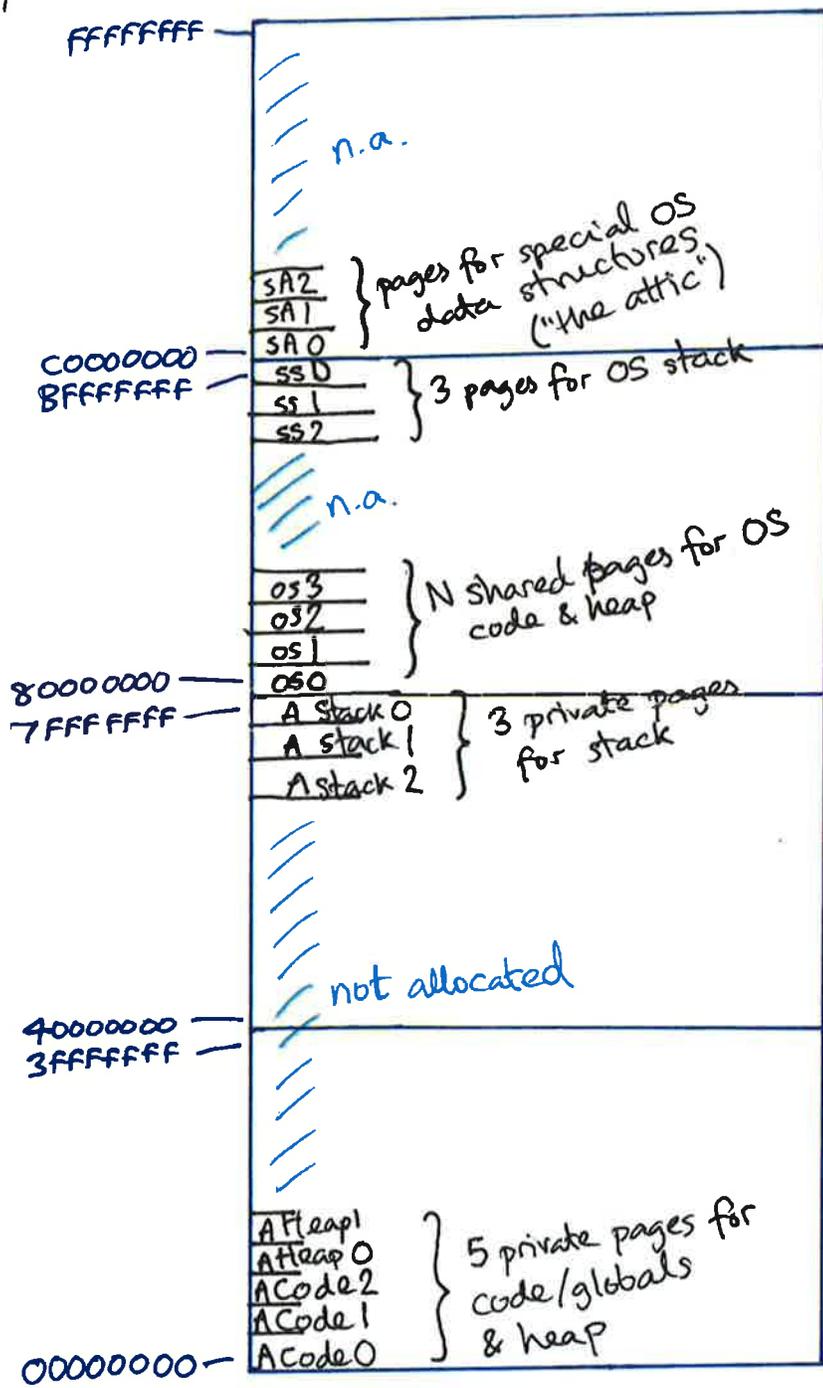
What's actually in memory -

2A000	Free
29000	OS attic page table
28000	B's stack page table
27000	Astack0
26000	Free
25000	Bstack0
24000	B's code/heap page table
23000	Free
22000	Acode1
21000	Bstack1
20000	Free
1F000	Bcode0
1E000	Free
1D000	B's page directory
1C000	OS stack page table
1B000	Aheap0
1A000	Free
19000	Astack1
18000	A's stack page table
17000	Bcode1
16000	Acode2
15000	A's code/heap page table
14000	Free
13000	Aheap1
12000	Astack2
11000	Bstack3
10000	Free
F000	Acode0
E000	Bstack2
D000	Bcode2
C000	OS code/heap page table
B000	A's page directory
A000	BHeap0
9000	SA2
8000	SA1
7000	SA0
6000	SS2
5000	SS1
4000	SS0
3000	OS3
00002000	OS2
00001000	OS1
00000000	OS0

OS's pages in use very likely these were allocated all at once, at startup

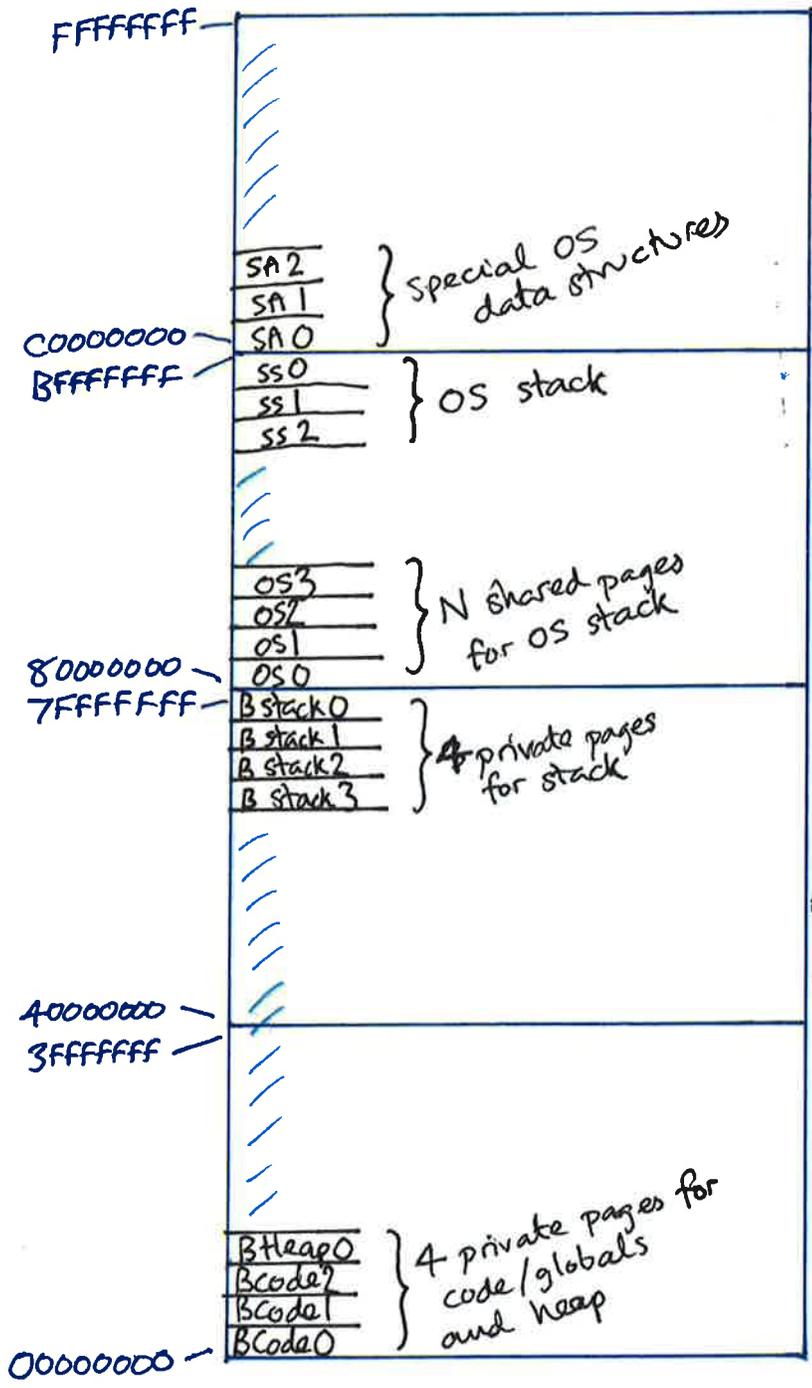
What's actually in memory -
Physical

Process A's view of the system



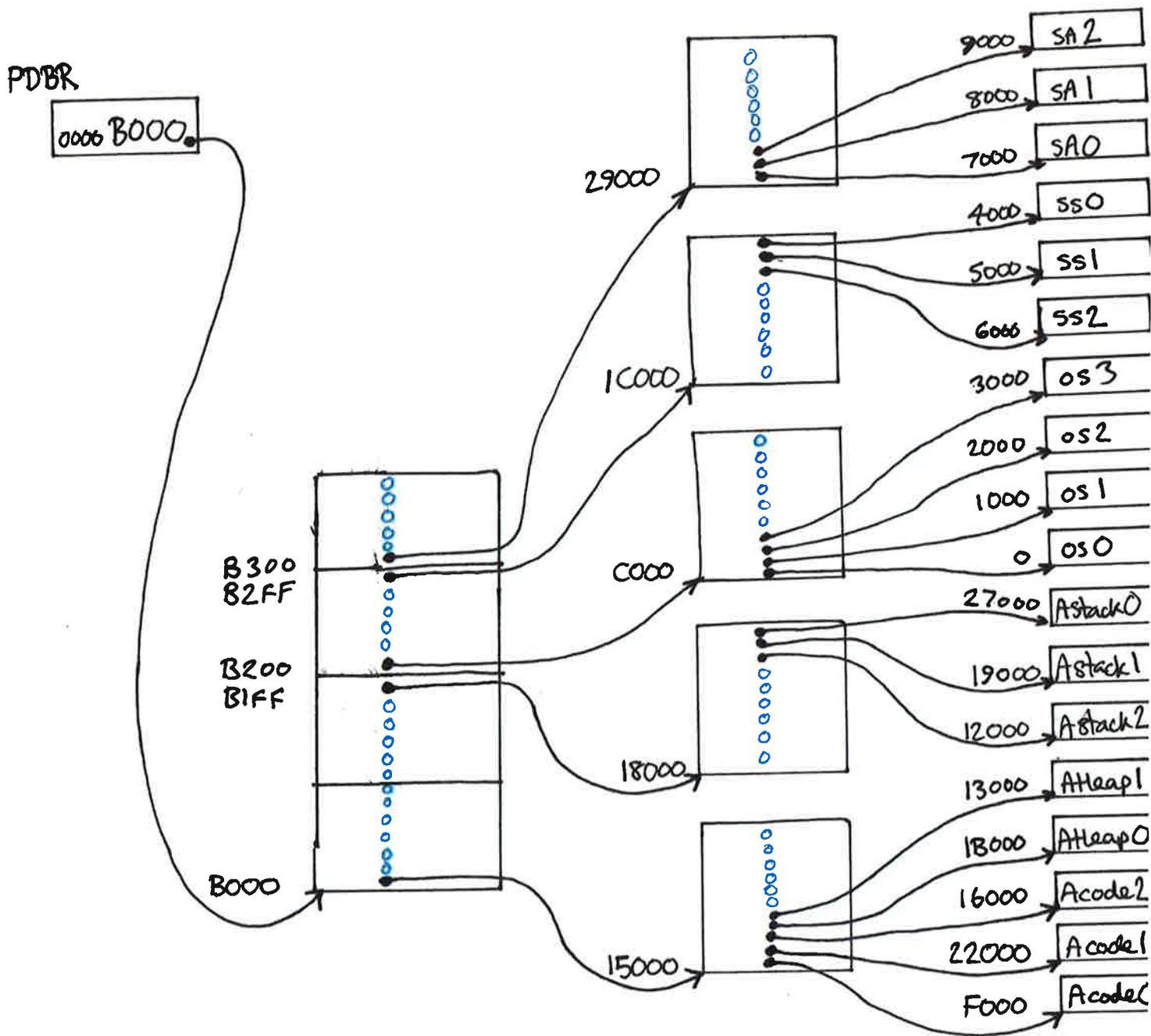
Process A's
view of the
system —
Virtual

process B's view of the system - virtual



Process B's view
of the system —
Virtual

page table structure when process A is running



Page table structure when process B is running

PDBR

0001D000

