Base & Bound registers

Two special privileged registers : `base` and `bound`
On each load/store/jump

- Physical address = virtual address + `base`
- Check 0 ≤ virtual address < `bound`, else trap to kernel

✓ OS can change these registers to move the process in memory

✓ OS must re-load base these register on context switch

Base + Bound Trade-offs

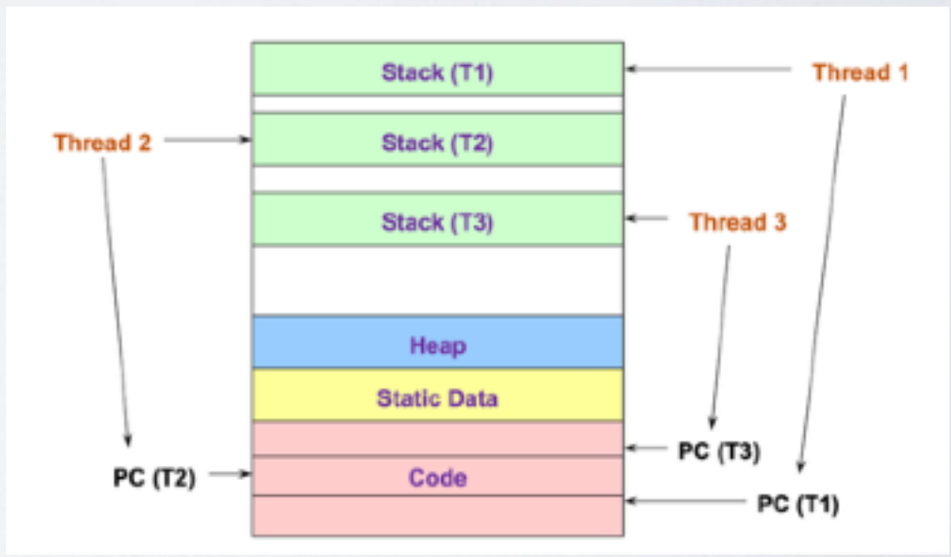
Advantages

✓ Cheap in terms of hardware : only two registers

✓ Cheap in terms of cycles : do add and compare in parallel

Disadvantages

- Growing a process is expensive
- No way to share code or data



➡ **Solution :** segmentation i.e separate code, stack and data segments