Segmentation
Maybe I shouldn't have brought the termites.
Overview

- History
- What is Paging?
- Paging Example
- Pros/Cons of Paging.
- What of Segmentation.
- Addressing Segments.
- Segmentation Example
- Pros/Cons of Segmentation.
- Video #1
Overview cont.

- Segmentation vs. Paging
- Segmentation with Paging
- Multics - quick case study
- Implementation: Segmentation w/paging: MULTICS
- Video 2
- Summary
- Class Questions
History

- Virtual memory was developed in approximately 1959 – 1962, at the University of Manchester for the Atlas Computer, completed in 1962.

- In 1961, Burroughs released the B5000, the first commercial computer with virtual memory based on segmentation.
What is Paging?

• During paging, the system divides the memory into pages.
• A page is a *physical* entity of a *fixed size*.
• Memory can be divided into large page sizes or small page sizes.
• A whole page must be used even if only holding a very small portion of data.
Paging example
Paging

Small Pages:
– Large amounts of data cannot be held on a single page.
– Pages are frequently swapped in and out causing page faults.
  • A page fault is what occurs when a page is too small to hold program
    – Program thrashing
  • Thrashing is when virtual memory is constant state of paging causing
    many exchanges of data in memory for data on disk.

Large Pages:
- Require less swapping
- Memory is quickly depleted
- Under utilization of physical memory due to fragmentation
Pros/Cons of Paging

Advantages:

● Efficient memory usage

● Simple partition management due to discontiguous loading and fixed partition size

● No compaction necessary

● Easy to share pages
Pros/Cons of Paging

Disadvantages:

- Job Size $\leq$ Memory Size
- Internal fragmentation (half the page size on the average)
- Need special hardware for address translation
- Some main memory space used for PMT's
- Address translation lengthens memory cycle times
Segmentation

A compromise was needed between large and small page sizes to maximize efficiency and improve performance.
Segmentation

- Memory-management scheme that break the main memory in logical pieces called “Segments”.
- Segments are divisions of computer memory of variable size.
- A program is a collection of segments. A segment is a logical unit such as:
  - main program, procedure, logical variables, global variables, symbols table, arrays…
What is Segmentation?

Virtual memory

MMU

offset < limit ?

no
memory access fault

yes

STBR
STLR

Segment Base + Offset physical address

0x00

External fragmentation

as in paging: valid, modified, protection, etc.

Physical memory

Seg 3 (stack)

Seg 1 (code)

Seg 2 (data)
Addressing Segments

- Let's first assume no paging in the system
- User generates logical addresses
- These addresses consist of a segment number and an offset into the segment
- Use segment number to index into a table
- Table contains the physical address of the start of the segment
  - often called the base address
- Add the offset to the base and generate the physical address
  - before doing this, check the offset against the limit
  - the limit is the size of the segment
Example of Segmentation

- subrountine
- stack
- symbol table
- main program

logical address space

<table>
<thead>
<tr>
<th>segment</th>
<th>limit</th>
<th>base</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>1000</td>
<td>1400</td>
</tr>
<tr>
<td>1</td>
<td>400</td>
<td>6300</td>
</tr>
<tr>
<td>2</td>
<td>400</td>
<td>4300</td>
</tr>
<tr>
<td>3</td>
<td>1100</td>
<td>3200</td>
</tr>
<tr>
<td>4</td>
<td>1000</td>
<td>4700</td>
</tr>
</tbody>
</table>

physical memory
Pros/ Cons of Segmentation

Advantages:

● No internal fragmentation **but** external fragmentation.
● May save memory if segments are very small and should not be combined into one page.
● Segment tables: only one entry per actual segment as opposed to one per page in VM
● Average segment size $>>$ average page size
Pros/ Cons of Segmentation

Disadvantages:
● External fragmentation
● Costly memory management algorithms
  - Segmentation: find free memory area big enough (search!).
  - Paging: keep list of free pages, any page is ok (take first!).
Video #1

GRE Computer Science Practice Book 2011

26. paging
fixed size
segmentation
allocated case-by-case
Paging vs. Segmentation

- Need the programmer be aware that this is being used?
- How many linear address spaces are there?
- Can the total address space exceed the size of physical memory?
- Is sharing of procedures between users facilitated?
Segmentation with paging

- Some modern processors allow usage of both, segmentation and paging alone or in a combination (Motorola 8030 and later, Intel 80386, 80486, Pentium) - the OS designers have a choice.
Segmentation with paging: MULTICS cont.

- The MULTICS OS Ran on Honeywell computers
- Segmentation + paging
- Up to 218 segments
- Segment length up to 216 36-bit words
- Each program has a segments table (itself a segment)
- Each segment has a page table
MULTICS data-structures
MULTICS memory reference procedure

1. Use segment number to find segment descriptor
   
   *Descriptor segment is itself paged because it may be large. The descriptor-base-register points to its page table*

2. Check if segment’s page table is in memory
   - if not a segment fault occurs
   - if there is a protection violation TRAP (fault)

3. page table entry examined, a page fault may occur
   - if page is in memory the start-of-page address is extracted from page table entry

4. offset is added to the page origin to construct main memory address

5. perform read/store etc.
MULTICS Address Translation Scheme

- Segment number (18 bits)
- Page number (6 bits)
- Page offset (10 bits)
### MULTICS TLB

<table>
<thead>
<tr>
<th>Comparison field</th>
<th>Segment number</th>
<th>Virtual page</th>
<th>Page frame</th>
<th>Protection</th>
<th>Age</th>
<th>Is this entry used?</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>4</td>
<td>1</td>
<td>7</td>
<td>Read/write</td>
<td>13</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>6</td>
<td>0</td>
<td>2</td>
<td>Read only</td>
<td>10</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>12</td>
<td>3</td>
<td>1</td>
<td>Read/write</td>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>1</td>
<td>0</td>
<td>Execute only</td>
<td>7</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>2</td>
<td>12</td>
<td>Execute only</td>
<td>9</td>
<td>1</td>
</tr>
</tbody>
</table>

- Simplified version of the MULTICS TLB
- Existence of 2 page sizes makes actual TLB more complicated
MULTICS Additional checks during segment link (call)

- Since segments are mapped to files, ACL (access control list) are checked with first access (open)
- Protection rings are called

A very advanced architecture for 1970’s!
Implementation
Segmentation with paging: MULTICS

Presenting the program ....

[Image of memory management and notepad windows]
Video 2
Summary / Conclusion

- What is Paging?
- What is Segmentation?
- Examples of Paging and Segmentation.
- Pros/Cons of Paging and Segmentation.
- Two Videos.
- Segmentation vs. Paging.
- Segmentation with Paging.
- Multics.
- Implementation Segmentation with Paging: MULTICS.
Questions?