Husain Alshehhi

Report

1. Changes in page_table.C
    1. Implementing register_vmpool: which takes a vm pool puts in in an array. So far, the register_vmpool array accespts at most 10 vmpools.
    2. Implementing freepage: which simply deletes the corresponding frame.
    3. Updaing handle_fault: which handles address virtually after enabling the page_table. It first checks if the address is valid (within the range of some vmpools) and then allocates a frame for the page or for the entry virtually.
2. Implementing vmpool: the idea is to allocates reagions of different sizes. Each region consists of two main information: (a) whether or not the region is available; (b) the exact size of the region.
    1. Vmpool constructor initalizes the variables.
    2. Allocate: creates region. Regions in this case are similar to a linked list: the end of each region is the beginning of another. When a request is made, allocates try to find the nearest big-enough region. If the region is too big, it splits it.
    3. Release: changes the availibility variable of a region. Combines regions if possible.
    4. Is_legitimate: checks if the address is within the boundaries.