ABSTRACT

Computer pioneers correctly predicted that programmers would want unlimited amounts of fast memory. Since fast memory is expensive, an economical solution to that desire is a memory hierarchy organized into several levels. The goal of the memory hierarchy is to provide a memory system with cost almost as low as the cheapest level of memory and speed almost as fast as the fastest level. Since the fast upper memory levels are small, programs with poor locality tend to access data from the lower levels of the memory hierarchy. Therefore, these programs run slower than programs with good locality. Given the technology and application trends, efficiently executing large applications on a hierarchy of memories and access to the slower disks to make the access compatible with the new emerging technologies, remains a challenge. Most of the emerging technologies today using embedded operating systems are planning to convert their operating system platforms to linux.

Moreover, applications that use large data sets frequently exhibit poor performance because the size of their working set exceeds the real memory, causing excessive page faults, and ultimately exhibit thrashing behavior.

In this project, we are mainly concerned with describing a memory compression solution to this problem that adapts the allocation of real memory between uncompressed and compressed pages. The key idea of using of compressed memory is to set aside part of main memory to hold compressed data. By compressing some of the data space, the effective memory size available to the applications is made larger allowing more and more pages of the application to be made available more frequently and excessive disk accesses are avoided. The system manages its resources dynamically based on the varying demands of each application and on the situational requirements that are data dependent. Hence, the technique has been modelled and designed to be implemented for modifying the memory module of the linux kernel. The proposed design is being implemented in Linux-2.6.21 kernel.