



FIGURE 7-2. Virtual machine memory layout

HOW ELSE COULD YOU DO IT?

The standard means of virtualizing virtual memory (when you cannot change the operating system) is to use *shadow page tables*. With these, the guest deals with pseudophysical addresses (i.e., contiguous and beginning at 0) in place of physical addresses.

The guest maintains its own page tables against this address space. However, they cannot be used by the hardware, because they don't correspond to real physical addresses. Therefore, the hypervisor monitors updates to these guest page tables, and uses them to construct a shadow page table, which translates virtual addresses to real physical addresses.

This method clearly incurs some overhead, but it is necessary when you cannot modify the operating system. Xen uses a variant of this method for hardware virtualized guests, as described later in this chapter.

Since Xen is not constrained by having to support unmodified operating systems, it is free to introduce virtual block and network drivers. Both operate in a similar manner: they comprise a frontend driver in the guest virtual machine and a backend driver in the virtualization software. The two devices communicate using a ring buffer, which is a high-performance mechanism for transferring large volumes of data between virtual machines. This results in a flexible layered architecture (Figure 7-3): the frontend implements the operating system's network or block device interface so that it appears to the operating system as a regular hardware device, and the backend connects the virtual device to real hardware. A virtual block device might be connected to a file containing a disk image or a real disk partition; a virtual