

What is claimed is:

1. A method operated by a service virtual machine, comprising
invoking, by a device model of the service virtual machine, a device driver of the service virtual machine to control a ~~part~~ virtual function interface of an input/output (I/O) device to implement an I/O operation by use of I/O information, which is related to the I/O operation and is written by a guest virtual machine;

wherein the device model, the device driver, and the ~~part~~ virtual function interface of the I/O device are assigned to the guest virtual machine; and

wherein the device model is configured to emulate a virtual I/O device or a physical I/O device.

2. The method of claim 1, further comprising if the ~~part~~ virtual function interface of the I/O device can not work compatibly with an architecture of the guest virtual machine, then:

translating, by the device driver, the I/O information complying with the architecture of the guest virtual machine into shadow I/O information complying with an architecture of the ~~part~~ virtual function interface of the I/O device; and

translating, by the device driver, updated shadow I/O information complying with the architecture of the ~~part~~ virtual function interface of the I/O device into updated I/O information complying with the architecture of the guest virtual machine, wherein the updated shadow I/O information was updated by the ~~part~~ virtual function interface of the I/O device in response to the implementation of the I/O operation.

3. The method of claim 1, further comprising:
maintaining, by the device driver, status of the ~~part~~ virtual function interface of the I/O device after the I/O operation is implemented.
4. The method of claim 1, further comprising;
informing, by the device model, the guest virtual machine that the I/O operation is implemented.
5. The method of claim 1, wherein the I/O information is written in a data structure starting from a head pointer that is controllable by the ~~part~~ virtual function interface of the I/O device.
6. The method of claim 1, wherein a tail pointer indicating end of I/O information is updated by the guest virtual machine.
7. An apparatus, comprising:
a device model and a device driver, wherein the device model invokes the device driver to control a ~~part~~ virtual function interface of an input/output (I/O) device to implement an I/O operation by use of I/O information which is related to the I/O operation and is written by a guest virtual machine, wherein the device model is configured to emulate a virtual I/O device or a physical I/O device, and wherein the device model, the device driver and the ~~part~~ virtual function interface of the I/O device are assigned to the guest virtual machine.
8. The apparatus of claim 7, wherein if the ~~part~~ virtual function interface of the I/O device can not work compatibly with an architecture of the guest virtual machine, then the device driver:

translates the I/O information complying with the architecture of the guest virtual machine into shadow I/O information complying with an architecture of the ~~part~~ virtual function interface of the I/O device; and

translates updated shadow I/O information complying with the architecture of the ~~part~~ virtual function interface of the I/O device into updated I/O information complying with the architecture of the guest virtual machine, wherein the updated shadow I/O information was updated by the ~~part~~ virtual function interface of the I/O device in response to the implementation of the I/O operation.

9. The apparatus of claim 7, wherein the device driver further maintains status of the ~~part~~ virtual function interface of the I/O device after the I/O operation is implemented.

10. The apparatus of claim 7, wherein the device model further informs the guest virtual machine that the I/O operation is implemented.

11. The apparatus of claim 7, wherein the I/O information is written in a data structure starting from a head pointer that is controllable by the ~~part~~ virtual function interface of the I/O device.

12. The apparatus of claim 7, wherein a tail pointer indicating end of I/O information is updated by the guest virtual machine.

13. A machine-readable medium, comprising a plurality of instructions which when executed result in a system:

invoking, by a device model of a service virtual machine, a device driver of the service virtual machine to control a ~~part~~ virtual function interface of an input/output (I/O) device to implement an I/O operation by use of I/O information, which is related to the I/O operation and is written by a guest virtual machine,

wherein the device model, the device driver and the ~~part~~ virtual function interface of the I/O device are assigned to the guest virtual machine; and

wherein the device model is configured to emulate a virtual I/O device or a physical I/O device.

14. The machine-readable medium of claim 13, wherein if the ~~part~~ virtual function interface of the I/O device can not work compatibly with an architecture of the guest virtual machine, then the plurality of instructions further result in the system:

translating, by the device driver, the I/O information complying with the architecture of the guest virtual machine into shadow I/O information complying with an architecture of the ~~part~~ virtual function interface of the I/O device; and

translating, by the device driver, updated shadow I/O information complying with the architecture of the ~~part~~ virtual function interface of the I/O device into updated I/O information complying with the architecture of the guest virtual machine, wherein the updated shadow I/O information was updated by the ~~part~~ virtual function interface of the I/O device in response to the implementation of the I/O operation.

15. The machine-readable medium of claim 13, wherein the plurality of instructions further result in the system:

maintaining, by the device driver, status of the ~~part~~ virtual function interface of the I/O device after the I/O operation is implemented.

16. The machine-readable medium of claim 13, wherein the plurality of instructions further result in the system:

informing, by the device model, the guest virtual machine that the I/O operation is implemented

17. The machine-readable medium of claim 13, wherein the I/O information is written in a data structure starting from a head pointer that is controllable by the ~~part~~ virtual function interface of the I/O device.

18. The machine-readable medium of claim 13, wherein a tail pointer indicating end of I/O information is updated by the guest virtual machine.

19. A system, comprising:
a hardware machine comprising an input/output (I/O) device; and
a virtual machine monitor to interface the hardware machine and a plurality of virtual machines, wherein the virtual machine comprises:

a guest virtual machine to write input/output (I/O) information related to an I/O operation; and

a service virtual machine comprising a device model and a device driver, wherein the device model invokes the device driver to control a ~~part~~ virtual function interface of the I/O device to implement the I/O operation by use of the I/O information, wherein the device model is configured to emulate a virtual I/O device or a physical I/O device, and wherein the device model, the device driver and the ~~part~~ virtual function interface of the I/O device are assigned to the guest virtual machine.

20. The system of claim 19, wherein if the ~~part~~ virtual function interface of the I/O device can not work compatibly with an architecture of the guest virtual machine, then the device driver of the service virtual machine further:

translates the I/O information complying with the architecture of the guest virtual machine into shadow I/O information complying with an architecture of the ~~part~~ virtual function interface of the I/O device; and

translates updated shadow I/O information complying with the architecture of the at least ~~part~~ virtual function interface of the I/O device into updated I/O information complying with the architecture of the guest virtual machine, wherein the updated shadow I/O information was updated by the ~~part~~ virtual function interface of the I/O device in response to the implementation of the I/O operation.

21. The system of claim 20, wherein the guest virtual machine writes the I/O information into a data structure starting from a head pointer which is updated by the ~~part~~ virtual function interface of the I/O device.

22. The system of claim 20, wherein the guest virtual machine updates a tail pointer indicating end of the I/O information.

23. The system of claim 20, wherein the virtual machine monitor transfers control of the system from the guest virtual machine to the service virtual machine, if detecting that the tail pointer is updated.

24. The system of claim 20, wherein the ~~part~~ virtual function interface of the I/O device updates the I/O information in response that the I/O operation is implemented.

25. The system of claim 20, wherein the device driver maintains status of the ~~part~~ virtual function interface of the I/O device after the I/O operation is implemented.

26. The system of claim 20, wherein the device model informs the guest virtual machine that the I/O operation is implemented.