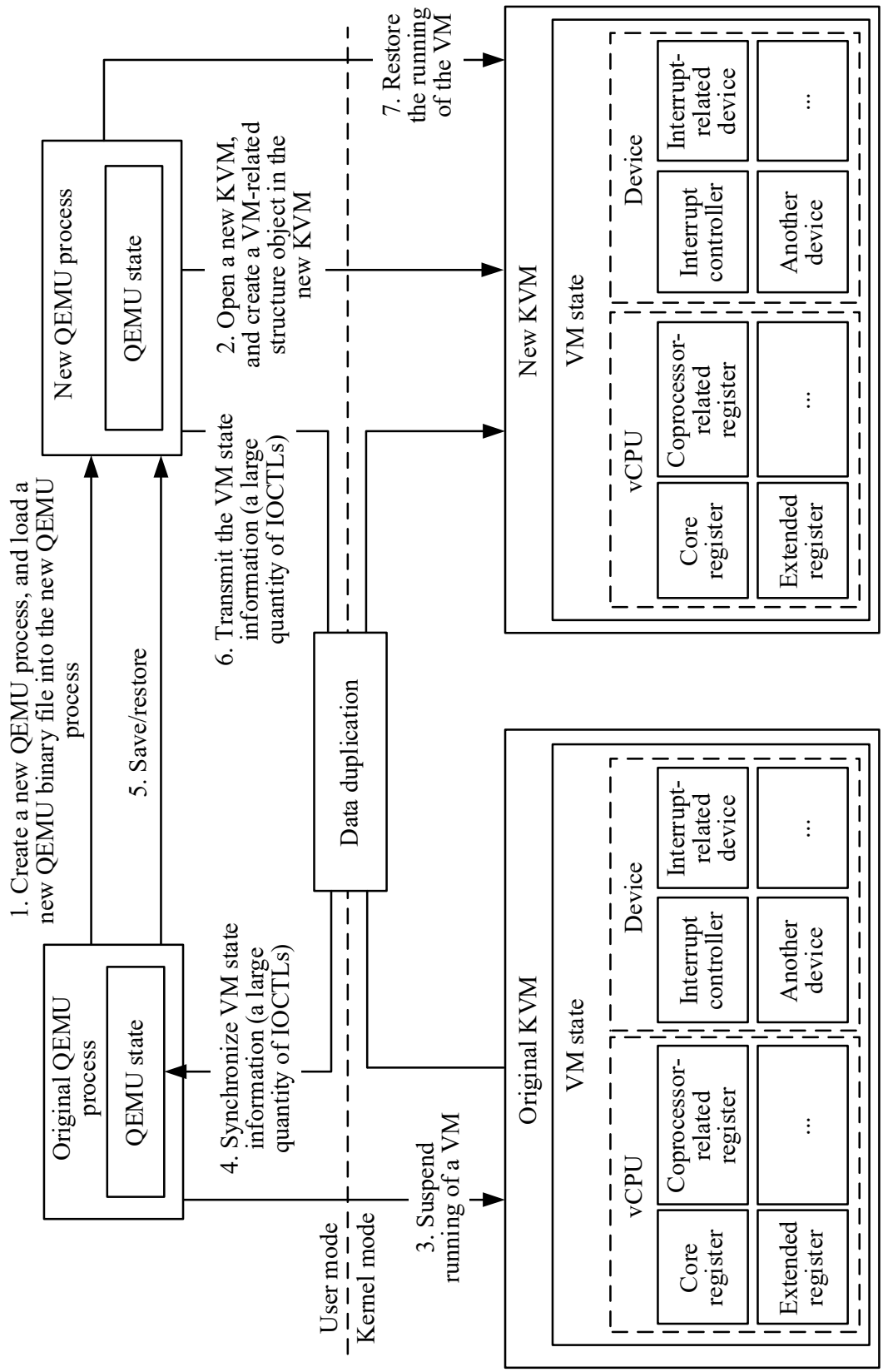

1. Create a new QEMU process, and load a new QEMU binary file into the new QEMU process
2. Open a new KVM, and create a VM-related structure object in the new KVM
3. Suspend running of a VM
4. Synchronize VM state information (a large quantity of IOCTLs)
5. Save/restore
6. Transmit the VM state information (a large quantity of IOCTLs)
7. Restore the running of the VM
Original QEMU process
QEMU state
New QEMU process
QEMU state
Data duplication
User mode
Kernel mode
Original KVM
VM state
vCPU
Core register
Coprocessor-related register
Extended register
...
Device
Interrupt controller
Another device
Interrupt-related device
...
New KVM
VM state
vCPU
Core register
Coprocessor-related register
Extended register
...
Device
Interrupt controller
Another device
Interrupt-related device
...

FIG. 1

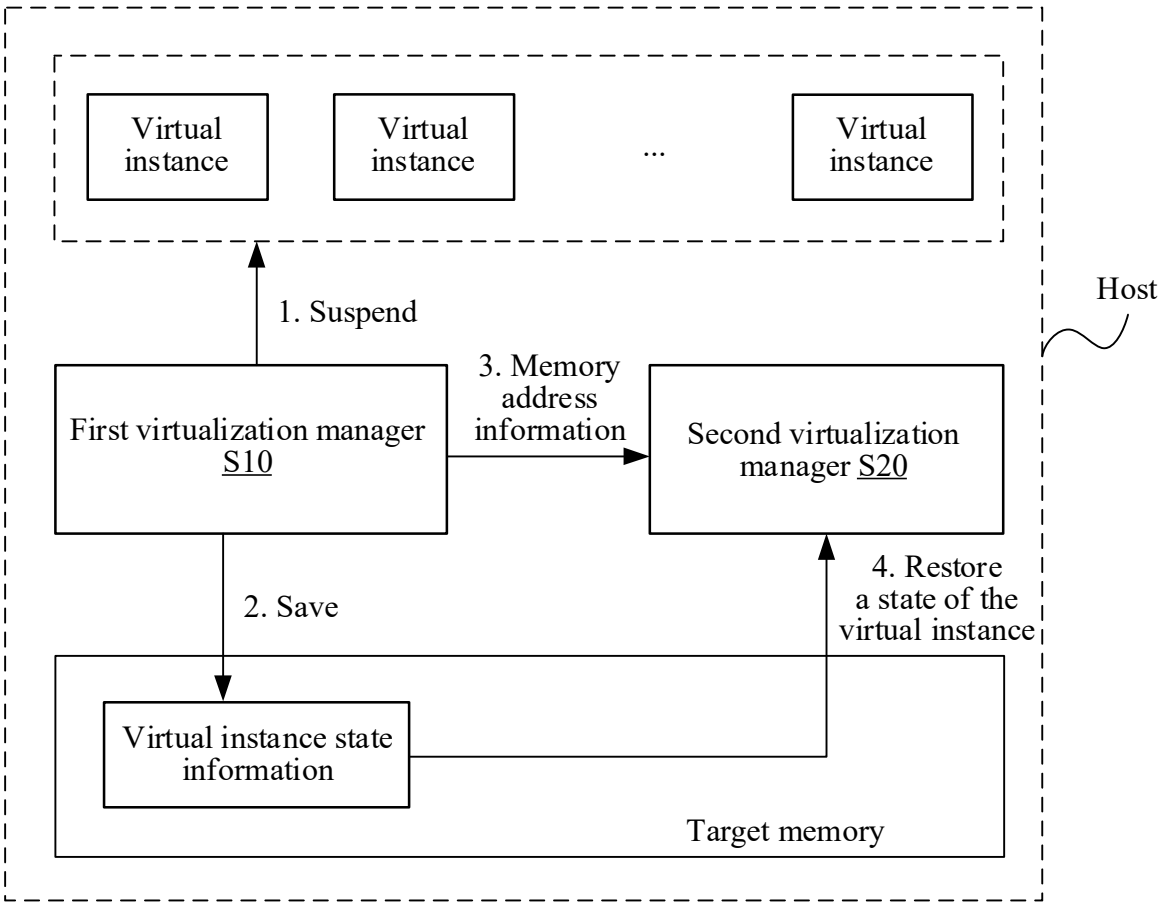

Virtual instance
Virtual instance
...
Virtual instance
Host
1. Suspend
First virtualization manager S10
3. Memory address information
Second virtualization manager S20
2. Save
4. Restore a state of the virtual instance
Virtual instance state information
Target memory

FIG. 2a

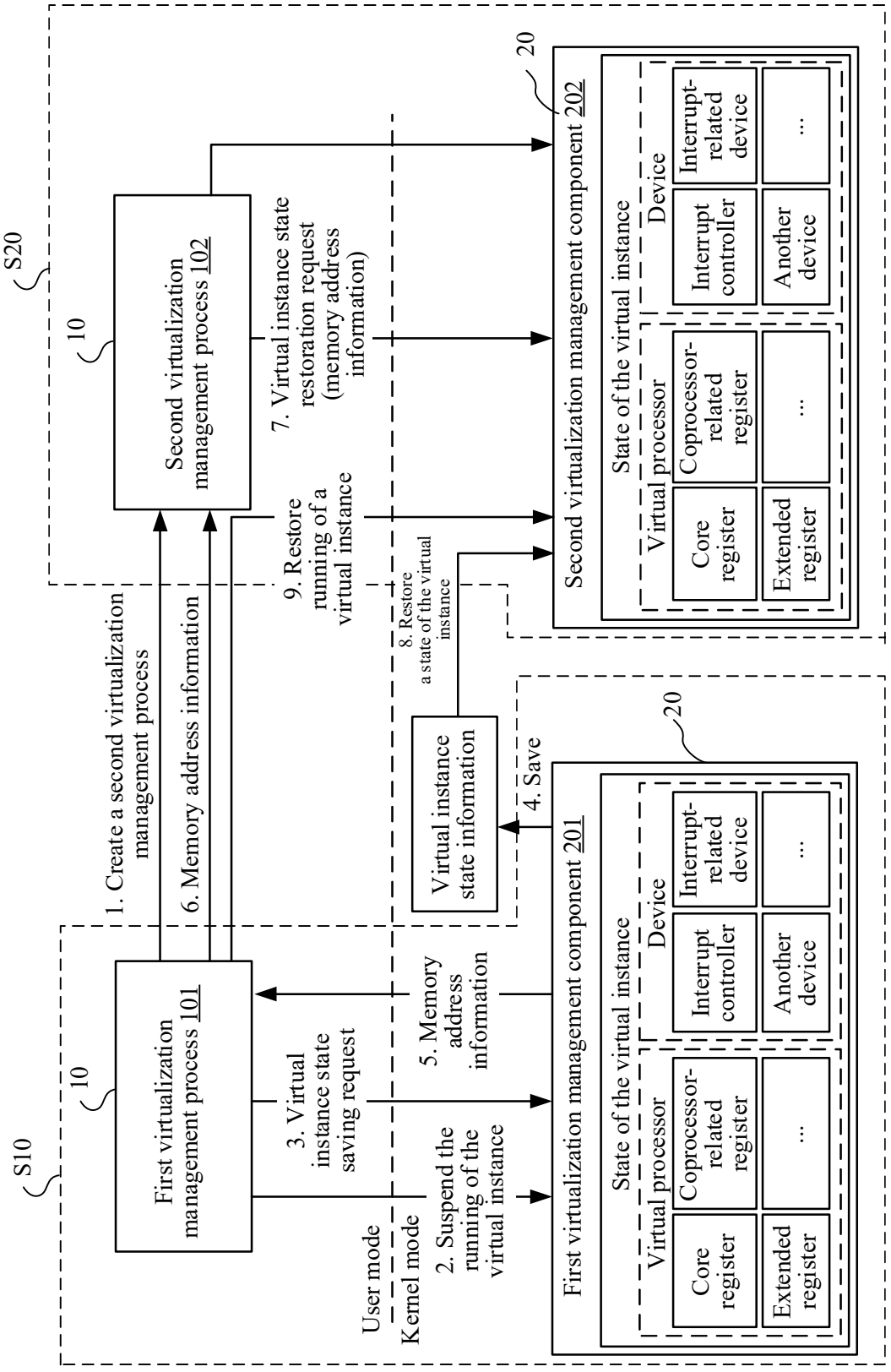

S10
S20
10
10
20
20
First virtualization management process 101
Second virtualization management process 102
1. Create a second virtualization management process
2. Suspend the running of the virtual instance
3. Virtual instance state saving request
4. Save
5. Memory address information
6. Memory address information
7. Virtual instance state restoration request (memory address information)
8. Restore a state of the virtual instance
9. Restore running of a virtual instance
Virtual instance state information
User mode
Kernel mode
First virtualization management component 201
Second virtualization management component 202
State of the virtual instance
Virtual processor
Core register
Extended register
Coprocessor-related register
...
Device
Interrupt controller
Another device
Interrupt-related device
...

FIG. 2b

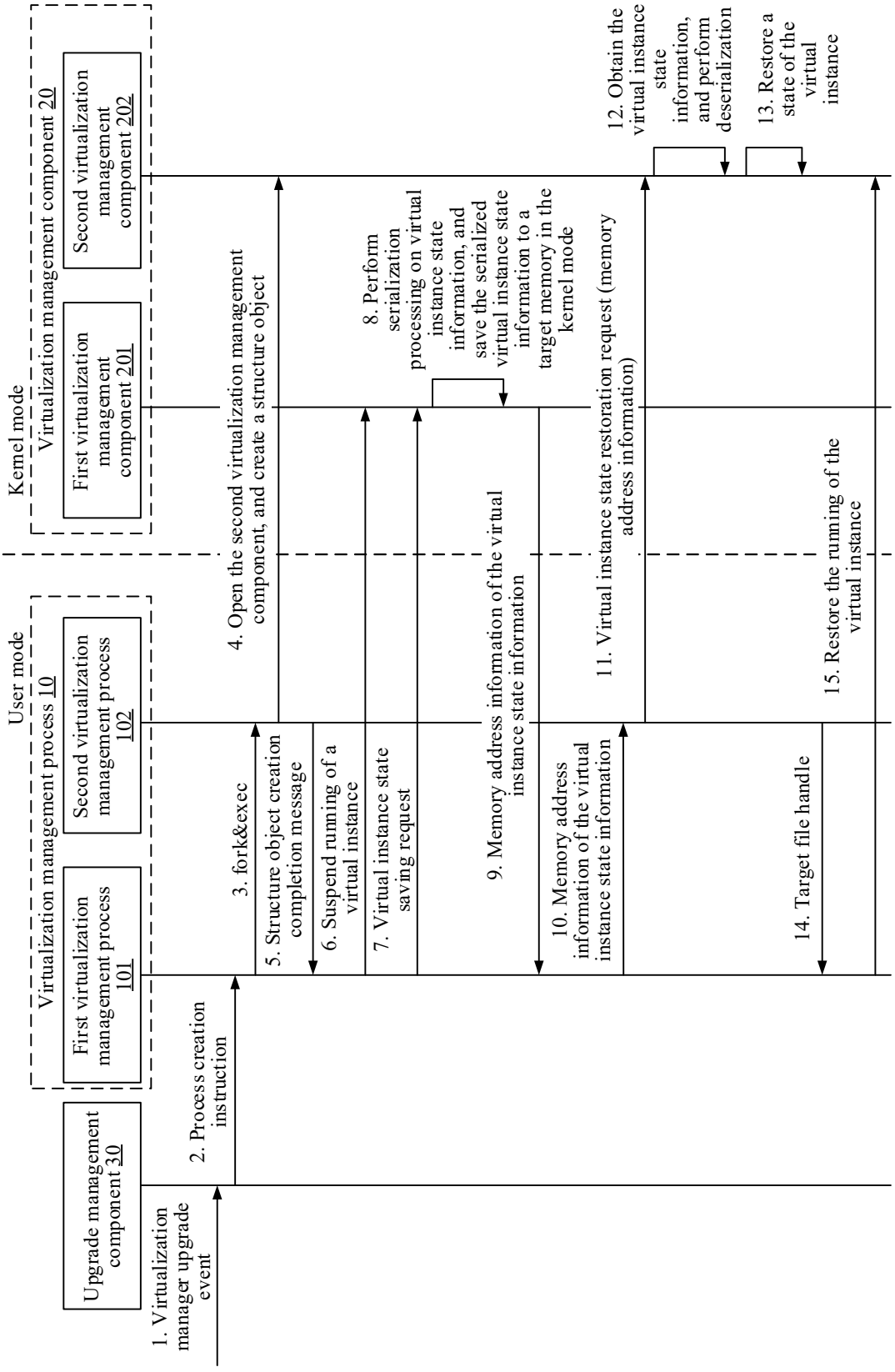

User mode
Kernel mode
Upgrade management component 30
Virtualization management process 10
First virtualization management process 101
Second virtualization management process 102
Virtualization management component 20
First virtualization management component 201
Second virtualization management component 202
1. Virtualization manager upgrade event
2. Process creation instruction
3. fork&exec
4. Open the second virtualization management component, and create a structure object
5. Structure object creation completion message
6. Suspend running of a virtual instance
7. Virtual instance state saving request
8. Perform serialization processing on virtual instance state information, and save the serialized virtual instance state information to a target memory in the kernel mode
9. Memory address information of the virtual instance state information
10. Memory address information of the virtual instance state information
11. Virtual instance state restoration request (memory address information)
12. Obtain the virtual instance state information, and perform deserialization
13. Restore a state of the virtual instance
14. Target file handle
15. Restore the running of the virtual instance

FIG. 3

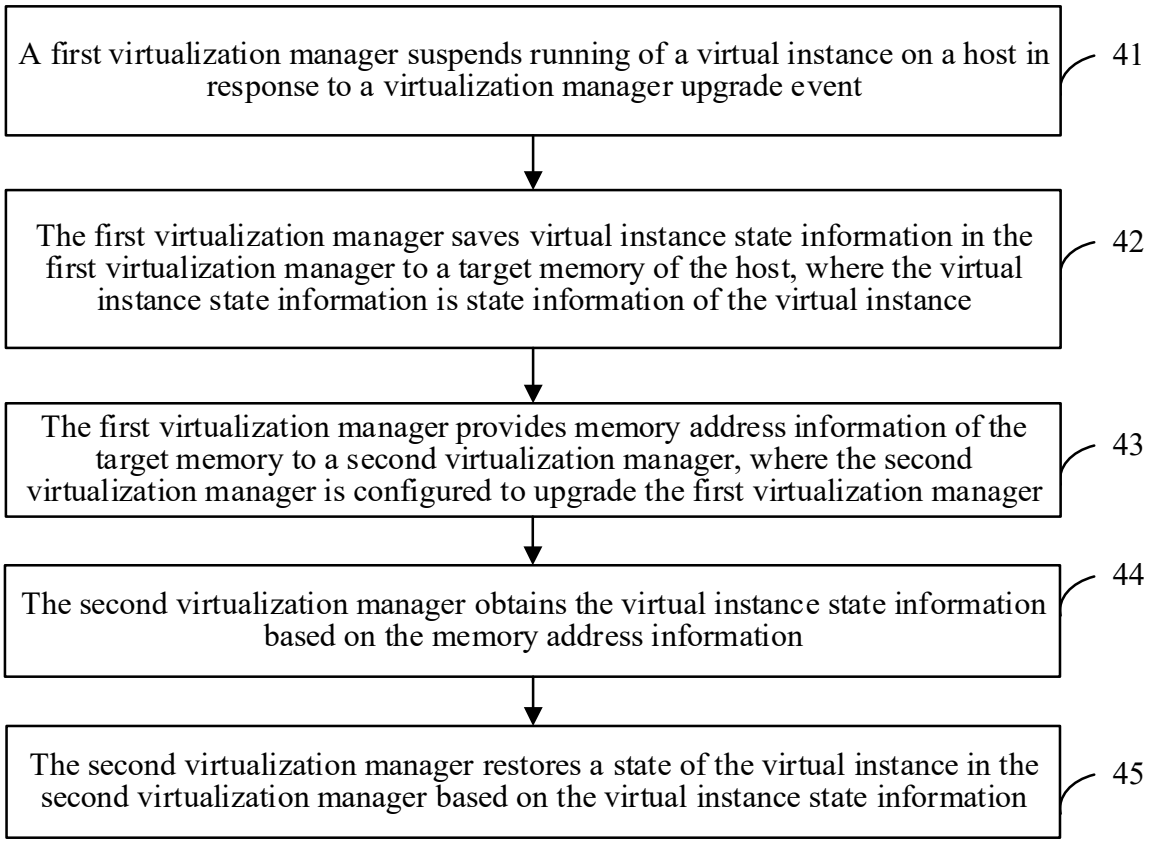

A first virtualization manager suspends running of a virtual instance on a host in response to a virtualization manager upgrade event
41
The first virtualization manager saves virtual instance state information in the first virtualization manager to a target memory of the host, where the virtual instance state information is state information of the virtual instance
42
The first virtualization manager provides memory address information of the target memory to a second virtualization manager, where the second virtualization manager is configured to upgrade the first virtualization manager
43
The second virtualization manager obtains the virtual instance state information based on the memory address information
44
The second virtualization manager restores a state of the virtual instance in the second virtualization manager based on the virtual instance state information
45

FIG. 4a

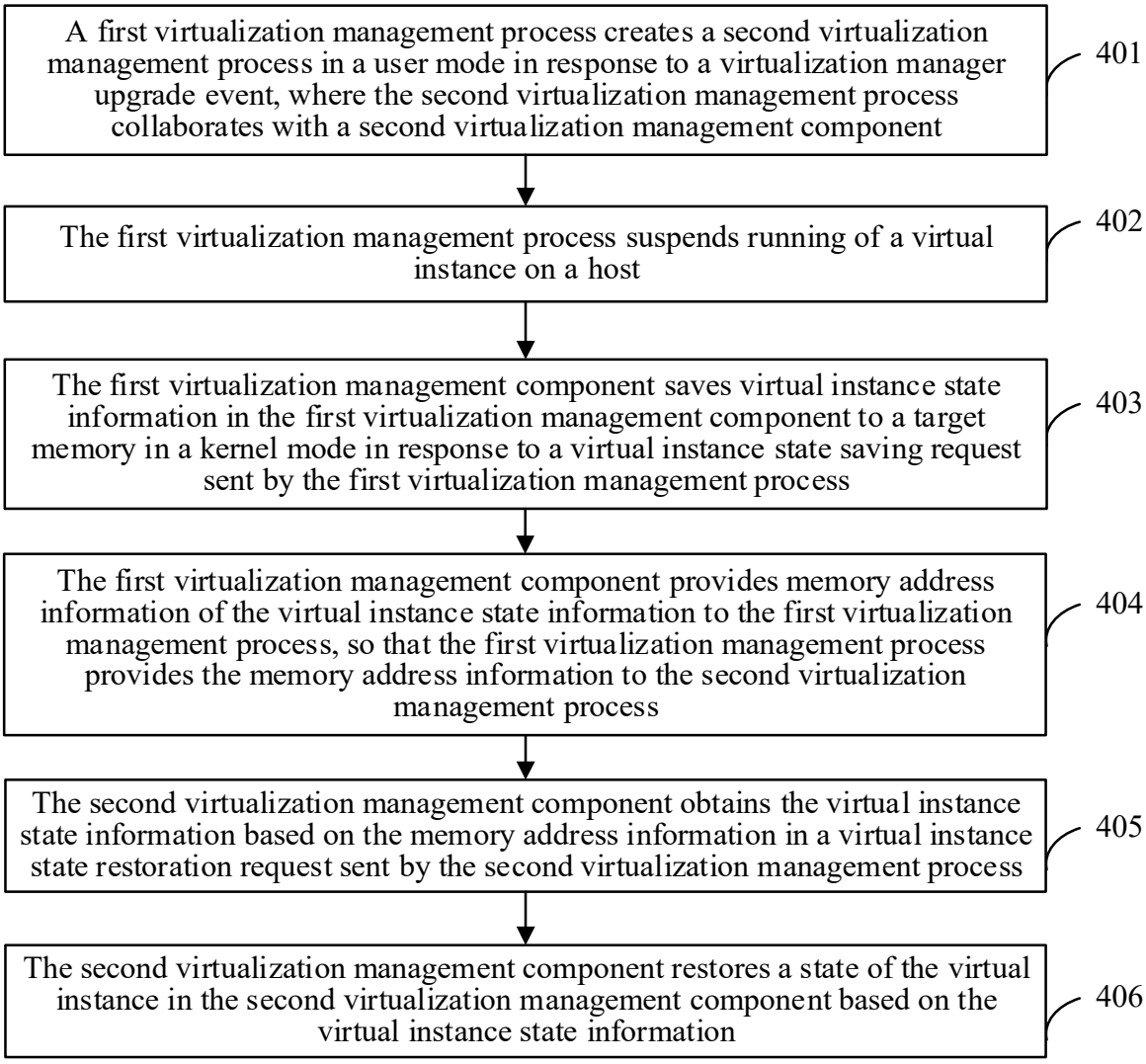

A first virtualization management process creates a second virtualization management process in a user mode in response to a virtualization manager upgrade event, where the second virtualization management process collaborates with a second virtualization management component
401
The first virtualization management process suspends running of a virtual instance on a host
402
The first virtualization management component saves virtual instance state information in the first virtualization management component to a target memory in a kernel mode in response to a virtual instance state saving request sent by the first virtualization management process
403
The first virtualization management component provides memory address information of the virtual instance state information to the first virtualization management process, so that the first virtualization management process provides the memory address information to the second virtualization management process
404
The second virtualization management component obtains the virtual instance state information based on the memory address information in a virtual instance state restoration request sent by the second virtualization management process
405
The second virtualization management component restores a state of the virtual instance in the second virtualization management component based on the virtual instance state information
406

FIG. 4b

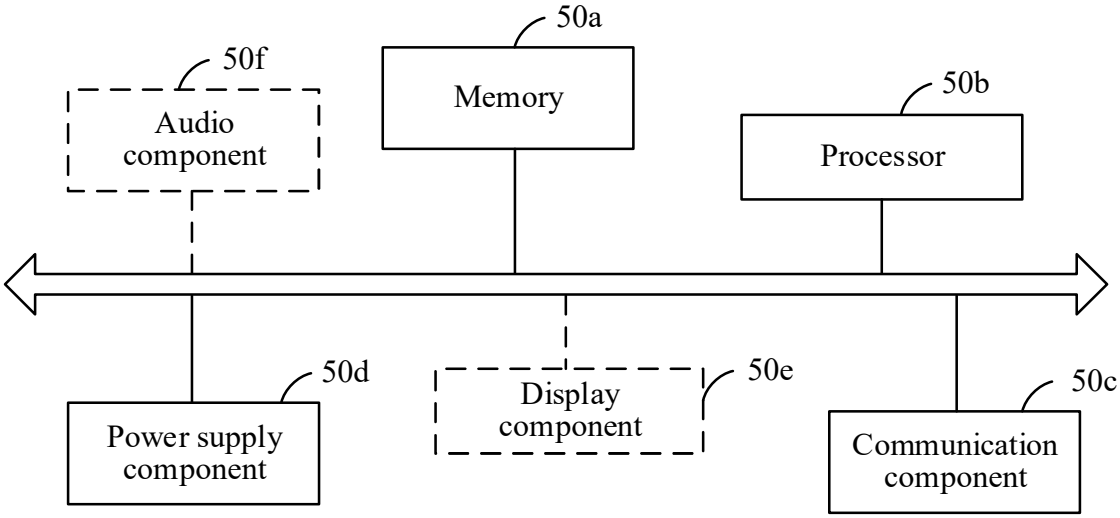

50a
Memory
50f
Audio component
50b
Processor
50d
Power supply component
Display component
50e
50c
Communication component

FIG. 5