HVM save/restore image format

1. summary

The paper describes a proposal of xen guest save/restore image format for supporting both HVM and PV guest.(red color indicate new added fields)

Whole pic of image format is as following:

Fields name	***Description***
Image header	General information for xen & guest
Memory image	Guest memory information
vcpu info	Guest vcpu context
HVM info	HVM guest specific information

2. image header

Image header provides general information of xen & guest including guest configuration, xen version number and physical cpu info.

Fields name	***Byte length***	***Description***
signature	14	A fixed string for check, currently “LinuxGuestRecord”, suggest “XenGuestRecord”
xen version	4	The xen source code version number when saving guest
image version	4	The image format version
Guest os type	4	Byte 3: processor type: 0-- ia32; 1--IPF; 2—PPC byte 2: sub type, if ia 32 processor: 0—32b; 1—pae; 2—em64t byte 1,0: reserved
Host os type	4	
cpu id	272	cpu id when saving guest. Intel processor has 17 input eax value and 16 byte result with each.
cpu freq	8	cpu frequency when saving guest
guest config length	4	The length of guest configuration(n)
guest config	n	all guest configuration including general config(e.g mem) and hvm specific config(e.g vnc, apic)

3. memory image

Guest memory image describe guest memory info and contents. Guest memory are divided into “batch” and we record how many pfns and their contents for each batch.

Fields name	***Byte length***	***Description***
num of pfns	4	Total number pfns for guest
num of pfns in batch 0	4	Number of pfns in this batch 0 (n)
pfn contents in batch 0	n X 4k	mem image contents in this batch 0. n X 4K on 32 bit host
num of pfns in batch 1	4	
pfn contents in batch 1	n X 4k	
...		

4 .vcpu info

vcpu's save/restore is different for PV&HVM smp guest. PV smp guest has only vcpu0's context, since other vcpu are hot unpluged when save. but HVM smp guest has more than one vcpu context.

Fields name	***Byte length***	***Description***
num of vcpus	4	Total number vcpus for guest
vcpu context leng	4	Length of the vcpu context (n)
vcpu context 0	n	#0 vcpu context (see 4.1)
vcpu context 1	n	
...		

4. 1 vcpu context

vmcs should be divided then incorporated into other vcpu context fields(user/control reg), but currently vcpu context for PV guest greatly differ from vmcs context. We postpone this task.

Fields name	***Description***
fpus	Guest fpus information
user regs	Guest user regs information
control regs	Guest control regs information
vmcs context	vmcs guest area for HVM guest

4.1.1 vmcs context

Fields name	***Byte length***	***Description***
valid flag	4	flag to indicate whether the following vmcs are valid. set it when save, unset it immediately after restore.

Fields name	***Byte length***	***Description***
vmcs guest area	sum of following fields	All vmcs fields needed to restore a hvm guest

4.1.1.1 vmcs guest area

For natural fields, we use 64 bit to support both 32/64 environment.

Fields name	***Byte length***	***Description***
eip	8	Execution pointer
esp	8	Stack pointer
eflags	8	Flags register
cr0	8	
cr3	8	
cr4	8	
idtr limit	4	idt information
idtr base	8	
gdtr limit	4	gdt information
gdtr base	8	
cs selector	4	cs information
cs limit	4	
cs base	8	
cs arbyte	4	
ds selector	4	ds information
ds limit	4	
ds base	8	
ds arbyte	4	
es selector	4	es information
es limit	4	
es base	8	
es arbyte	4	
ss selector	4	ss information
ss limit	4	
ss base	8	
ss arbyte	4	
fs selector	4	fs information
fs limit	4	
fs base	8	

Fields name	***Byte length***	***Description***
fs arbyte	4	
gs selector	4	gs information
gs limit	4	
gs base	8	
gs arbyte	4	
tr selector	4	task register information
tr limit	4	
tr base	8	
tr arbyte	4	
ldtr selector	4	ldtr information
ldtr limit	4	
ldtr base	8	
ldtr arbyte	4	
sysenter cs	4	
sysenter esp	8	
sysenter eip	8	
vir apic page address	8	Physical addr of 4KB virtual-APIC page that contains TPR shadow
TPR threshhold	4	Control TPR shadow fall
msr_items	48	some msr's info used for em64t HVM guest
cpu_state	8	cpu's extra state, e.g PAE, long mode enabled...

5. HVM information

This is HVM guest specific information in both hypervisor(HV) and device model(DM). The device model in HV and DM has same format.

Fields name	***Description***
HVM context	HVM guest information in HV
Device Model context	HVM guest information in DM

5.1 HVM context

hvm context is a long buffer shared between HV and control panel(CP) to transfer data, which is transparent to CP. HV=>CP when save and CP=>HV when restore.

Fields name	***Byte length***	***Description***
HVM magic number	4	Magic number for HVM guest
HVM version	4	HVM version when save

Fields name	*Byte length*	*Description*
HVM Context len	4	Fixed length for the long buffer, (n)
HVM buffer	n	The buffer contents

5.1.1 HVM buffer

Fields name	*Byte length*	*Description*
Device state 0	...	State of device 0 in HV
Device state 1	...	
....		

5.2 device model context

Fields name	*Byte length*	*Description*
DM signature	21	"QemuDeviceModelRecord”
Magic number	4	Magic number for Device model state
DM version	4	Version of device model state
Device state 0	...	State of device 0 in device model
Device state 1	...	
......		

5.2.1 device state format

This format apply for devices in both HV and CP

Fields name	*Byte length*	*Description*
idstr len	4	The length of idstr (n)
idstr	n	Idstr for this device, e.g. “xen i8254” or “ne2000”
instance id	4	The id number for same type device. e.g slave/master pic has same idstr, but different instance id
version id	4	The version for this device implementation
Record size	4	The device state size (m)
Record data	m	Device state contents