



Open Source
Technology
Center

Xen Support for Intel® LaGrande Technology

Joseph Cihula
Intel Corp.

Xen Summit
September 7-8, 2006



LaGrande Technology Overview

Removes BIOS and bootloader from trust chain

- Creates dynamic root of trust (DRTM)

HW-based measured launch

- Does not require VT

DMA protection

Reset memory protection

Safer Mode Extensions (SMX)

- LT processor instructions



LT Technology Enabling Platform (TEP)

Full LT support

- NoDMA table for DMA protection

Commercially available in Q4

Targeting academics and security developers

- For research and prototyping, not for production deployment

Xen code will be reference LT code

- This is why it is BSD licensed

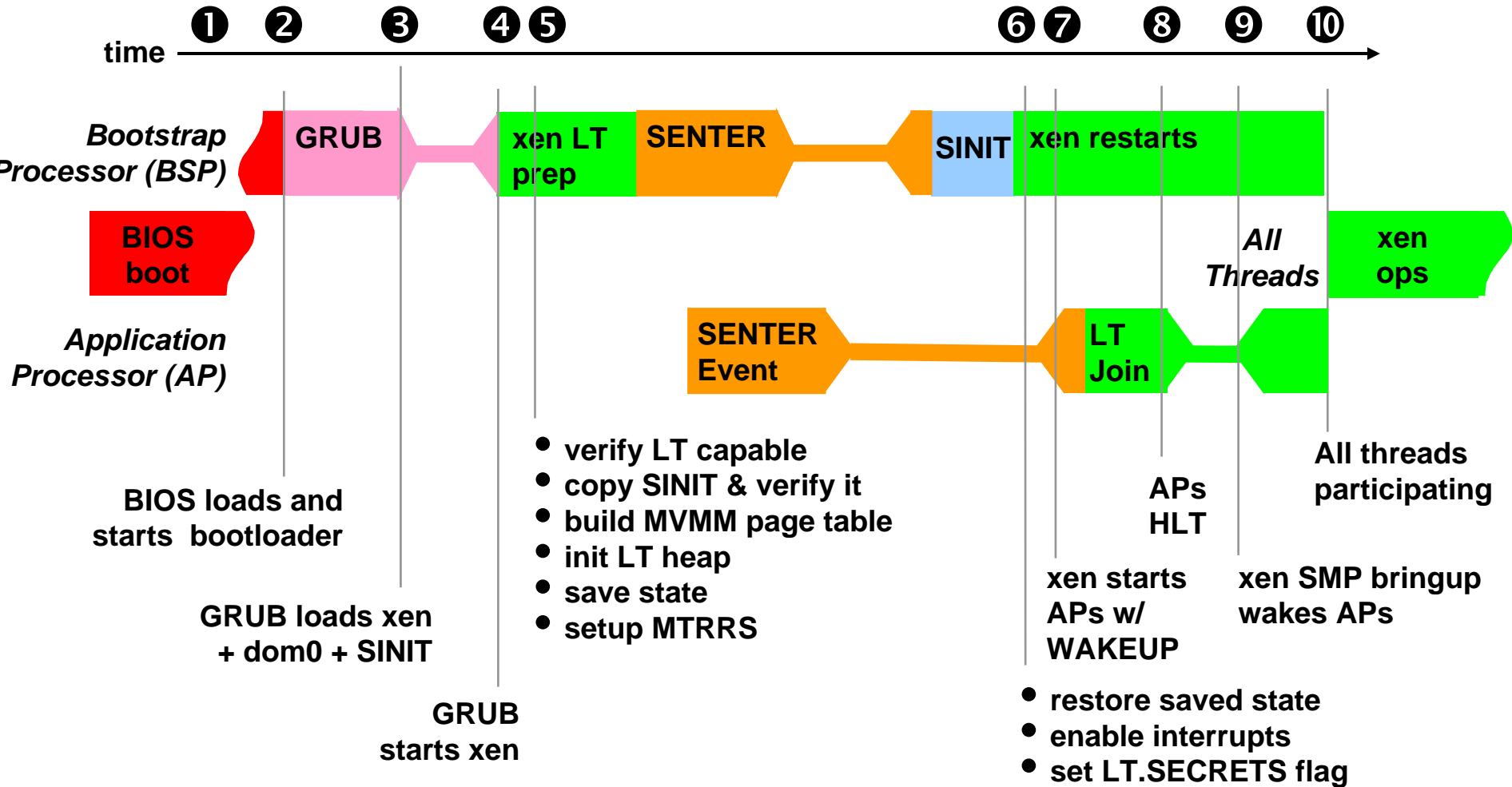


LT Integration into Xen

Three stages of LT integration:

- Xen launch
- Environment management
- Xen shutdown

Xen Launch Timeline



Environment Management

Protecting memory

- Xen must disallow privileged domain (I/O) access to LT private config space
 - Prevent mappings
- Xen must “disallow” access to TPM locality 2
 - Access to localities 3,4 are prevented by HW
 - “Disallow” = reads return 0xFF and writes are dropped
 - Can’t simply remove/prevent mappings because this generates faults
 - Must map locality 3 physical pages to locality 2 addr space

Adjusting memory map

- Xen uses e820 table from BIOS to determine RAM regions and MMIO space
 - This can’t be trusted in protected environment
- SINIT creates list of memory ranges and their types:
 - RAM “safe” for VMs, SMM areas, PCI-e config space, LT MMIO
- Need to adjust the e820 table:
 - Delete any ranges that don’t map to “safe” RAM
 - Mark LT data areas (heap, SINIT, NoDMA) as unavailable



LT Environment Teardown

LT process is largely reverse of measured launch

Xen process:

- All VMs are destroyed and MVMM begins shutdown
- Scrub all processors' states
- BSP should scrub memory that might have sensitive data
- Cap TPM dynamic PCRs
 - Extend with fixed value (e.g. 0xff's)
 - Prevents next environment from using them to unseal secrets
- Clear LT.SECRETS flag
- Close LT private config space
- Shutdown VT (VMXOFF)
- Call GETSEC[SEXIT]
- Perform reset/poweroff



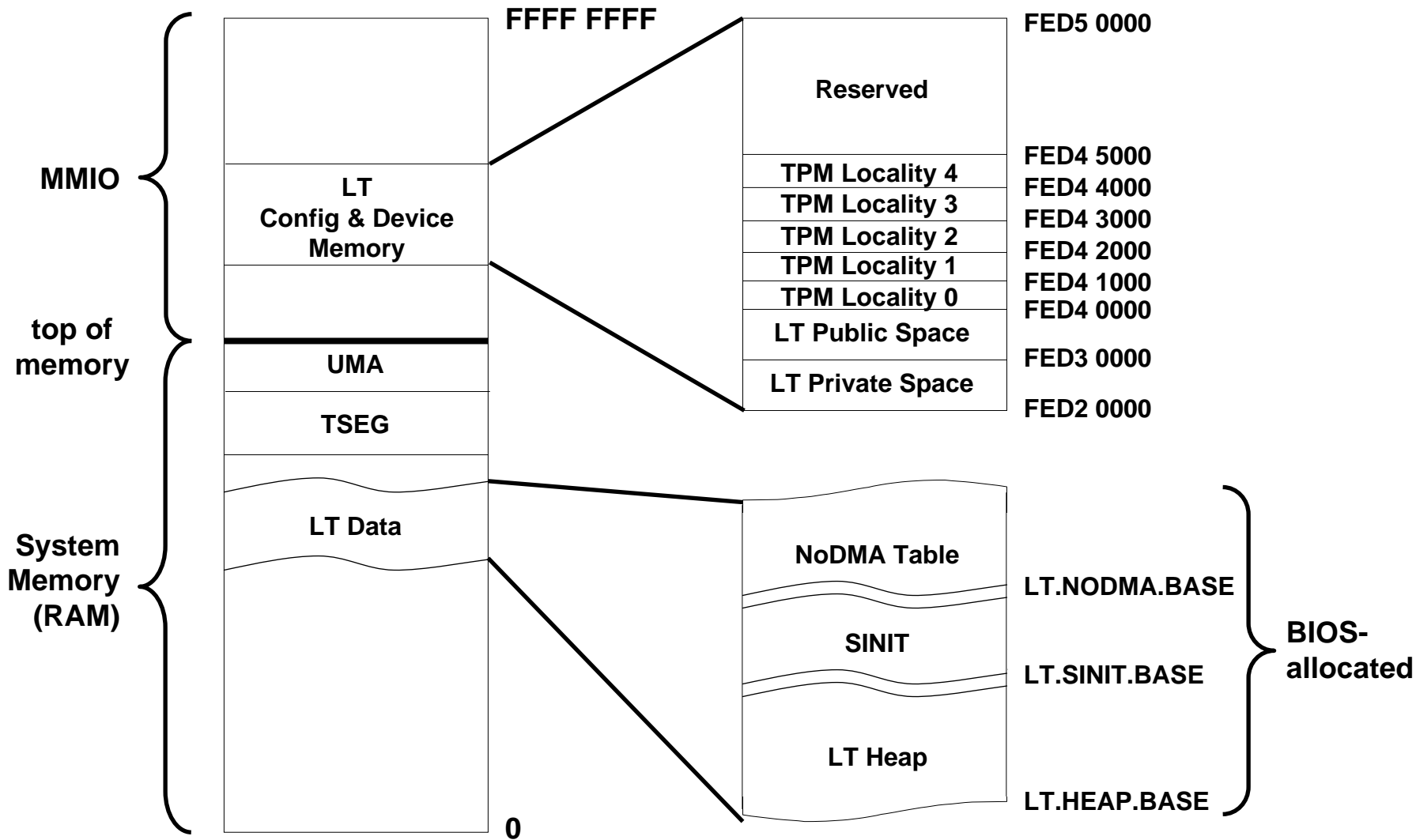
Remaining Tasks

- 64bit support
- SMP support
- Measurement of complete TCB
- Cap dynamic TPM PCRs on teardown
- Protect TPM locality 2
- DMA protection

Backup



LT Memory Layout



Processor State after SENTER

Processor state	ILP	RLP
CR0	Clear PG, AM, WP	Clear PG, CD, NW, AM, WP. Set PE, NE.
CR4	00004000H	00004000H
EFLAGS	00000002H	00000002H
EIP	AC.base (EBX) + [EntryPoint]	[LT.MVMM.JOIN+12]
EBP	AC.base (EBX)	NC
CS	Sel = [SegSel], base = 0, limit = FFFFFFFH, G = 1, D = 1, AR = 9BH	Sel = [LT.MVMM.JOIN + 8], base = 0, limit = FFFFFFFH, G = 1, D = 1, AR = 9BH
DS	Sel = [SegSel] + 8, base = 0, limit = FFFFFFFH, G = 1, D = 1, AR = 93H	Sel = [LT.MVMM.JOIN + 8] + 8, base = 0, limit = FFFFFFFH, G = 1, D = 1, AR = 93H
ES	Sel = [SegSel] + 8, base = 0, limit = FFFFFFFH, G = 1, D = 1, AR = 93H	Sel = [LT.MVMM.JOIN + 8] + 8, base = 0, limit = FFFFFFFH, G = 1, D = 1, AR = 93H
SS	Sel = [SegSel] + 8, base = 0, limit = FFFFFFFH, G = 1, D = 1, AR = 93H	Sel = [LT.MVMM.JOIN + 8] + 8, base = 0, limit = FFFFFFFh, G = 1, D = 1, AR = 93H
GDTR	Base = AC.base (EBX) + [GDTBasePtr], Limit = [GDTLimit]	Base = [LT.MVMM.JOIN + 4], Limit = [LT.MVMM.JOIN]
DR7	00000400H	00000400H
IA32_DEBUGCTL	0	0
IA32_EFER	0	0
IA32_MISC_ENABLE MSR	See Table 6.	See Table 6.
Performance counters and counter control	0	0