TDX Live Migration

Presented by: Wei Wang <wei.w.wang@intel.com>

Contributors: Isaku Yamahata, Ravi Sahita, Jun Nakajima, Dror Caspi, Jiewen Yao, Haidong Xia

intel®

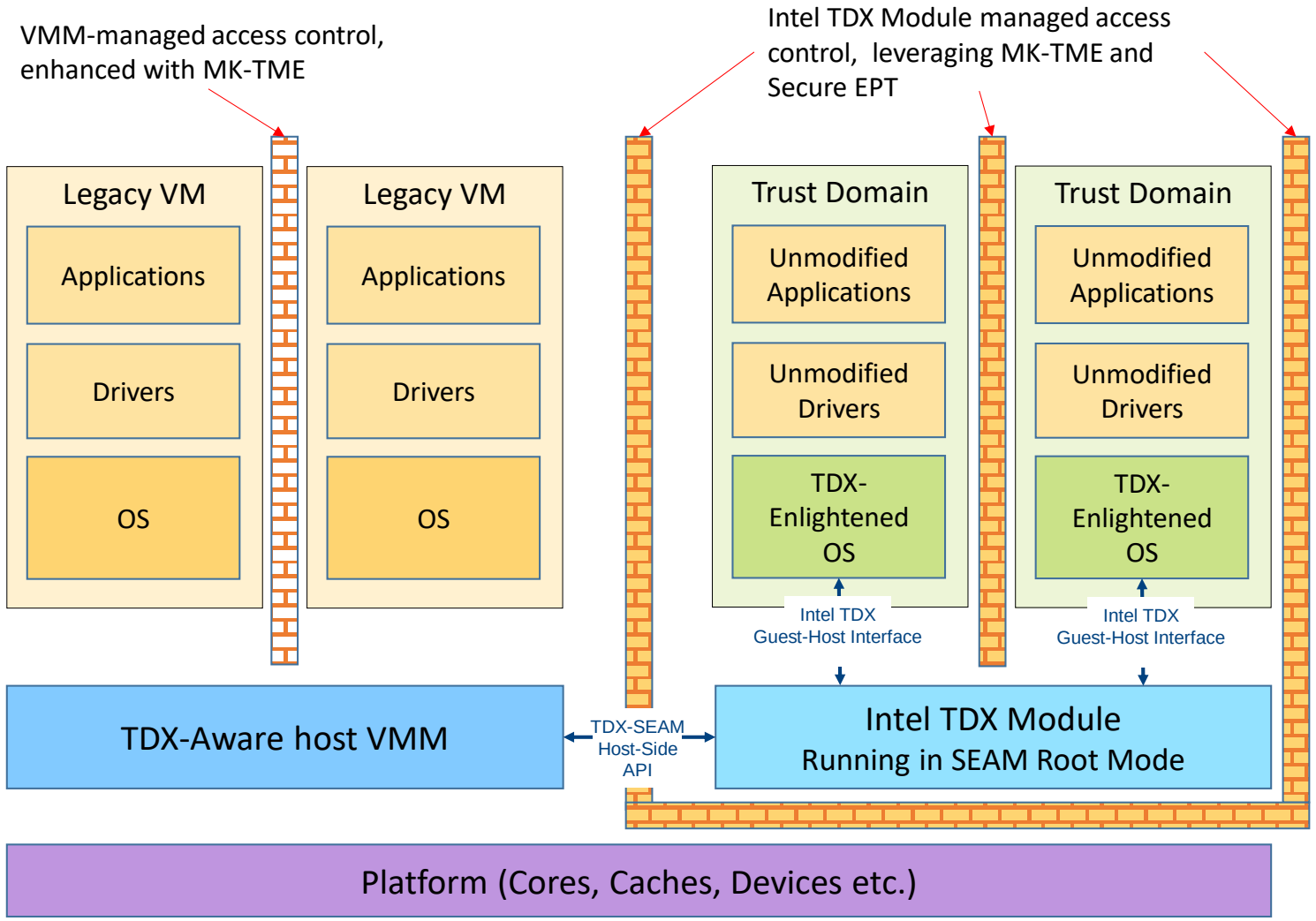
Agenda

- Background Introduction
- TDX Live Migration
- Initial PoC Results
- Status and Plan

Background Introduction

TDX Review

- Intel TDX Module runs in SEAM root mode to manage guest private states
- QEMU/KVM is removed from TCB
  - TD shared memory remains accessible
  - TD private memory is non-accessible
  - TD vCPU states are non-accessible
- KVM manages physical resources and assists TDX Module to virtualize TD via SEAMCALLs
  - E.g. allocate and offer pages to TDX module to build TD's secure EPT



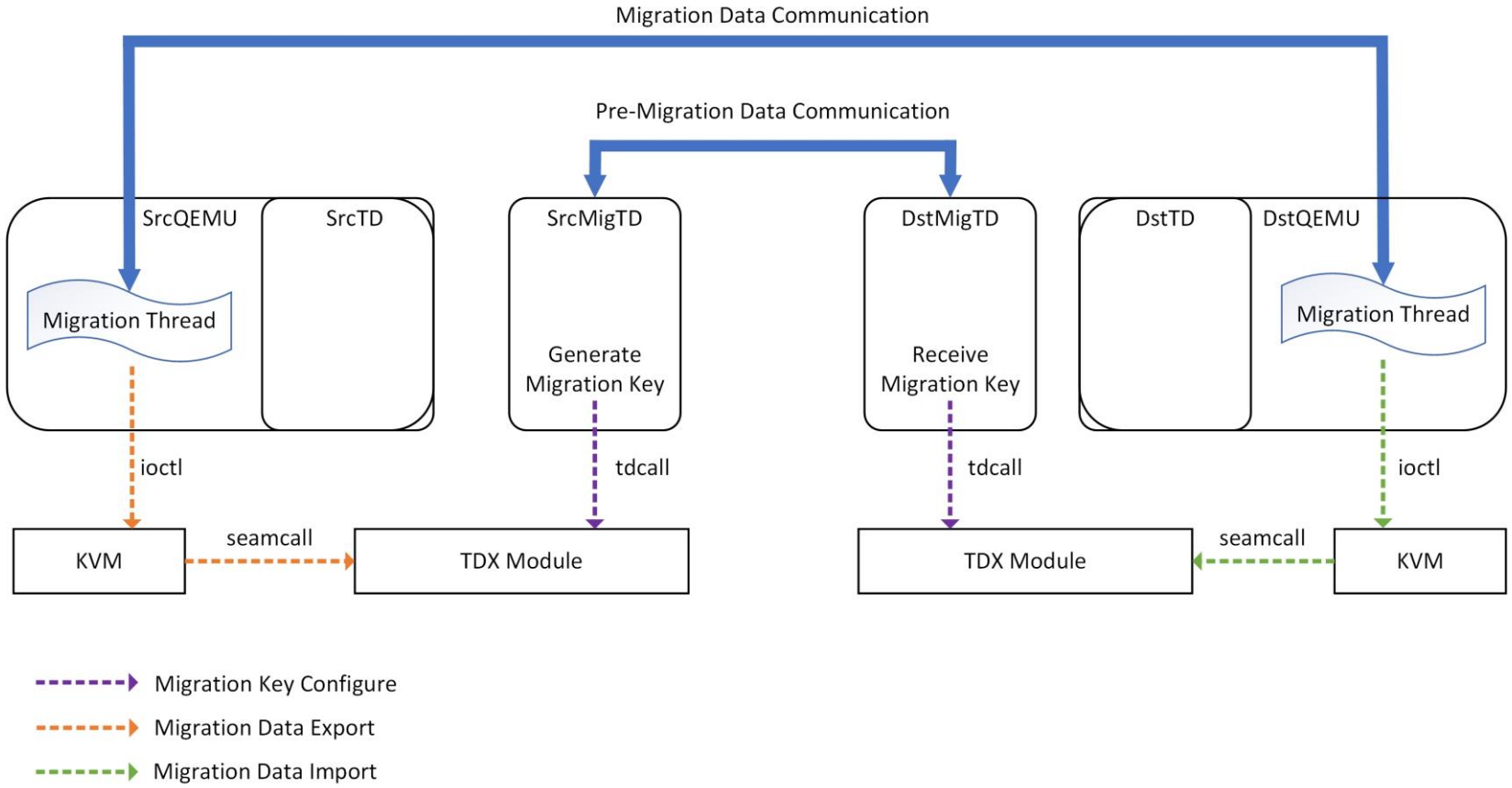
TDX Live Migration Callouts

- Dirty page logging
  - PML isn't supported to log dirty private pages in the first release
  - Seamcall to TDX Module to do write-protection on private pages
- Guest memory copy
  - QEMU doesn't have access to TD private pages
    - Seamcall to TDX module to export/import TD private pages with encryption/decryption
  - SEPTs on the destination need to be set up before importing a TD private page
- Huge page split
  - Not needed for the first release as TD works with 4KB pages only in the first place
- A common framework to abstract TDX migration implementations into the vendor specific layer

TDX Live Migration

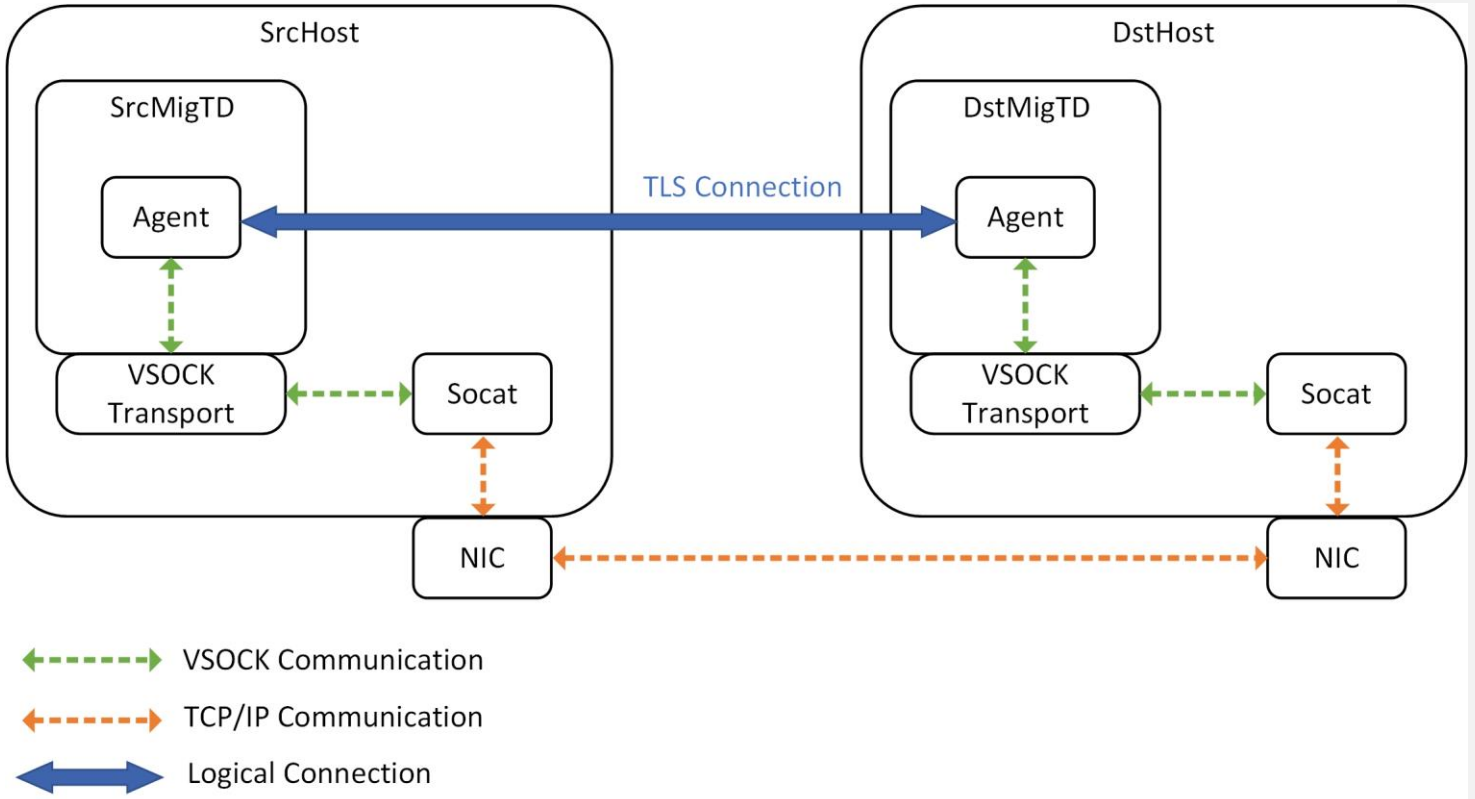
Bird's-eye View

- Pre-migration
  - Migration policy evaluation
    - Compatibility check
    - Security attestation
  - Migration key setup
    - Generated by SrcMigTD and securely transferred to DstMigTD
    - Set to TDX Module on both sides
- Migration data
  - States encrypted/decrypted by TDX Module using the migration key
    - TD private memory states
    - vCPU states
    - TD-scope states
  - States in clear texts
    - TD shared memory states



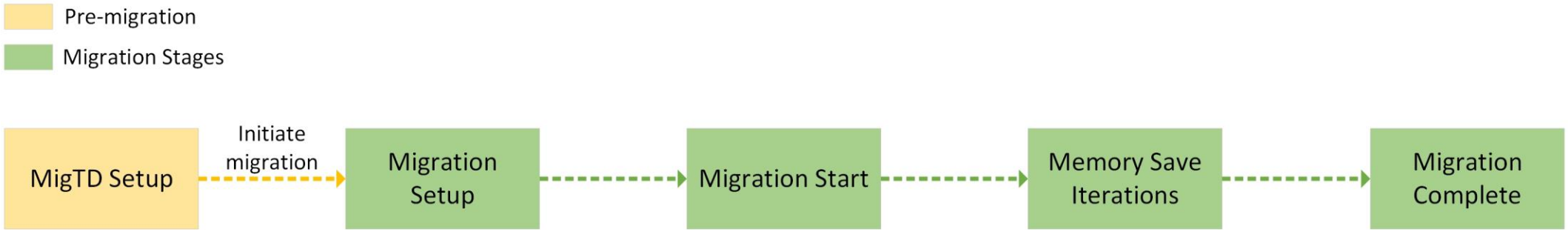
MigTD

- A service TD to assist the migration of guest TDs
  - Perform migration policy evaluation and migration key setup
  - Talk to TDX Module using TDCALLs
  - No interaction with the guest TD
  - VMM binds it to the guest TD that it assists using Seamcalls
  - One MigTD can assist the migration of multiple guest TDs at the same time
  - Part of the platform TCB, and included in the TD attestation
- MigTD communication
  - TLS connection between the source and destination MigTDs to keep the info exchange (e.g. migration key) secure
  - Use virtio-vsock or TDG.VP.VMCALL based VSOCK transport for Guest-Host communication
  - Socat to relay messages from guest to host network
- MigTD is vendor specific
  - Intel provides a reference design and RUST-based implementation, and cloud vendors can design on their own



Migration Flow

- KVM maintains a per kvm_memory_slot bitmap to indicate if a page is private or shared
  - Bits set/cleared upon EPT violations
  - Private pages go through the export/import steps
  - Shared pages go though the legacy migration path



- VMM boots a MigTD
- VMM binds MigTD to the guest TD: TDH.SERVTD.BIND
- MigTD generates a migration key and sets to TDX module: TDG.SERVTD.WR

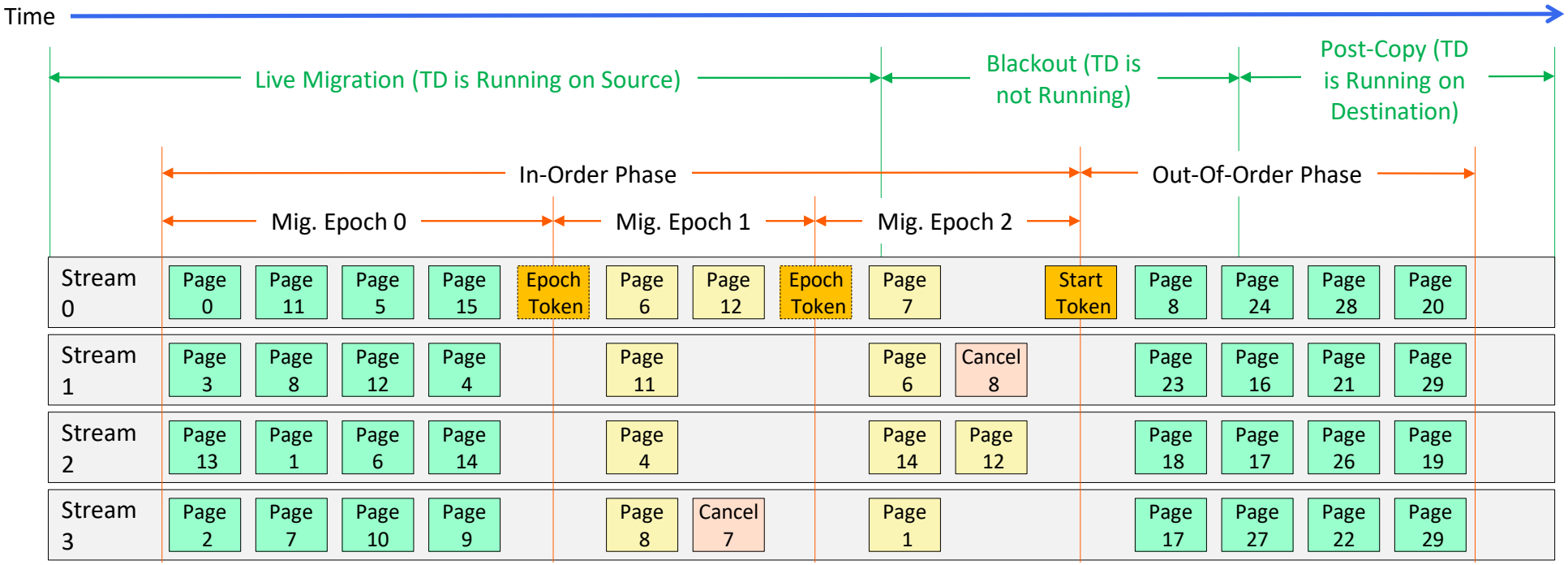
- Create one or multiple migration streams: TDH.MIG.STREAM.CREATE
- Start dirty page logging
- Huge page split
- Write-protection: TDH.EXPORT.BLOCKW TDH.EXPORT.UNBLOCKW

- Export TD-scope Immutable states: TDH.EXPORT.STATE. IMMUTABLE

- Export memory pages: TDH.EXPORT.MEM
- Mark the end at each round by exporting a token: TDH.EXPORT.TRACK

- Pause the guest TD: TDH.EXPORT.PAUSE
- Export the remaining memory pages
- Export mutable TD-scope states: TDH.EXPORT.STATE.TD
- Export vCPU states: TDH.EXPORT.STATE.VP
- Generate a start token: TDH.EXPORT.TRACK

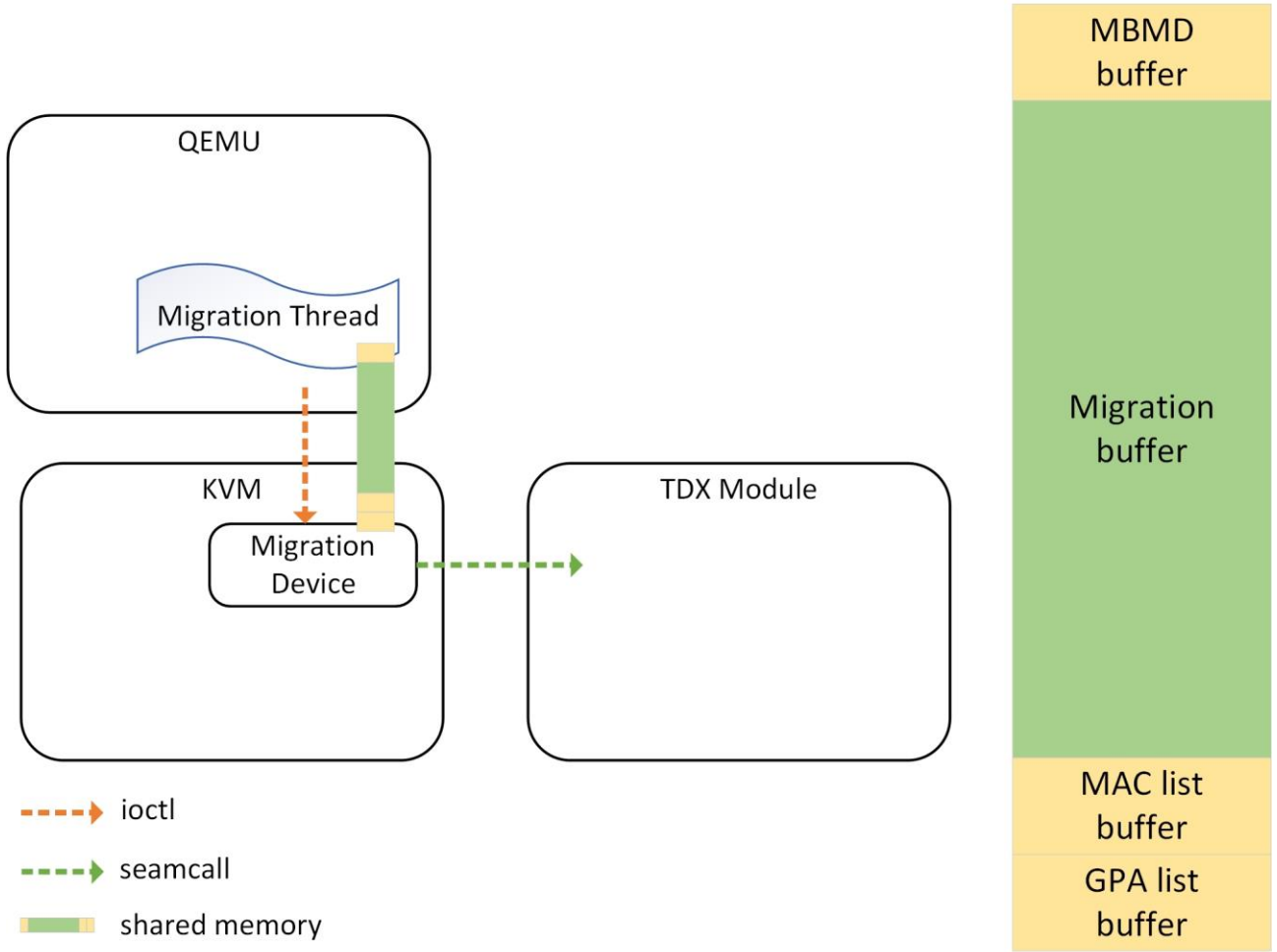
Migration Flow Con't



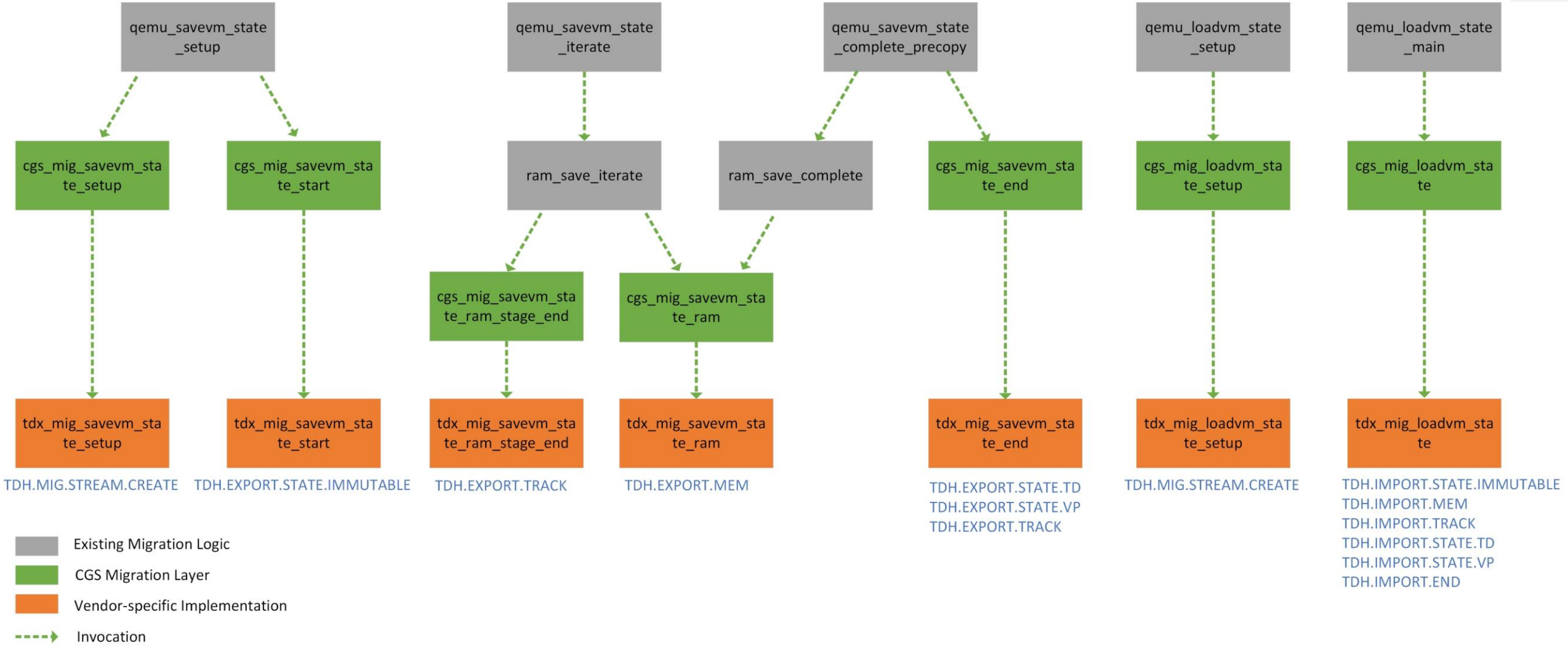
- In-Order Phase
  - Source TD is still running
  - Newer version of a page must be imported after the older version of this page has been imported in each round
    - QEMU naturally supports it, as each page gets migrated only once in each round
- Out-of-Order Phase
  - Source TD is paused
  - Used by post-copy, which will be supported later

Migration Data Transport

- A migration stream creates a migration device emulated via KVM device
  - QEMU migration thread ioctls on the device fd to send requests, e.g. export states
  - KVM device allocates a piece of memory mapped by the migration thread to transport the exported states
    - The memory is also given to TDX Module to export/import the encrypted states
- Shared Memory
  - MBMD buffer stores the migration bundle metadata
  - Migration buffer stores the exported or imported states
  - Mac list buffer stores a list of MACs corresponding to the TD private pages in the migration buffer
  - GPA list buffer stores a list of GPA entries corresponding to the TD private pages in the migration buffer
- Multifd supports multiple migration streams, so multilpe migration devices are created in KVM
  - Each device shares a piece of memory with its mutifd iothread



Confidential Guest Migration Framework



Initial PoC Results

Note: Results are from tests of legacy VM live migration with adding the estimated TDX overhead to memcpy (named pseudo-tdx)

Test Environment

- Testbed
  - CPU: Intel(R) Xeon(R) CPU E5-2699 v4 @ 2.20GHz
  - DRAM: DDR4, 2666MHZ
  - NIC: Intel 10-Gigabit X540-AT2
    - Direct cable connection on source and destination's NICs
- Live migration
  - Downtime: 300 ms (default)
  - Network bandwidth: No limit (i.e. maximum 10G)
- Legacy Guest
  - 8 vCPUs, 32GB RAM
  - No compression, but 0 page optimization is used
- Legacy Guest without 0 page optimization
  - 8 vCPUs, 32GB RAM
  - No compression and no 0 page optimization
- TD Guest, labelled Pseudo-TDX-xxxx
  - 8 vCPUs, 32GB RAM
  - No compression and no 0 page optimization
  - Modelled by adding extra xxxx cycles overhead memory read on SRC and write on DST
    - 2300 cycles = 0.24 * 4096 + 1000 additional transition latency + 300 syscall latency
    - 4000 cycles = 0.63 * 4096 + 1000 additional transition latency + 300 syscall latency
  - Pseudo-TDX-xxxx-multifd: multifd is enabled, with 4 channels (i.e. i/o threads) to send data

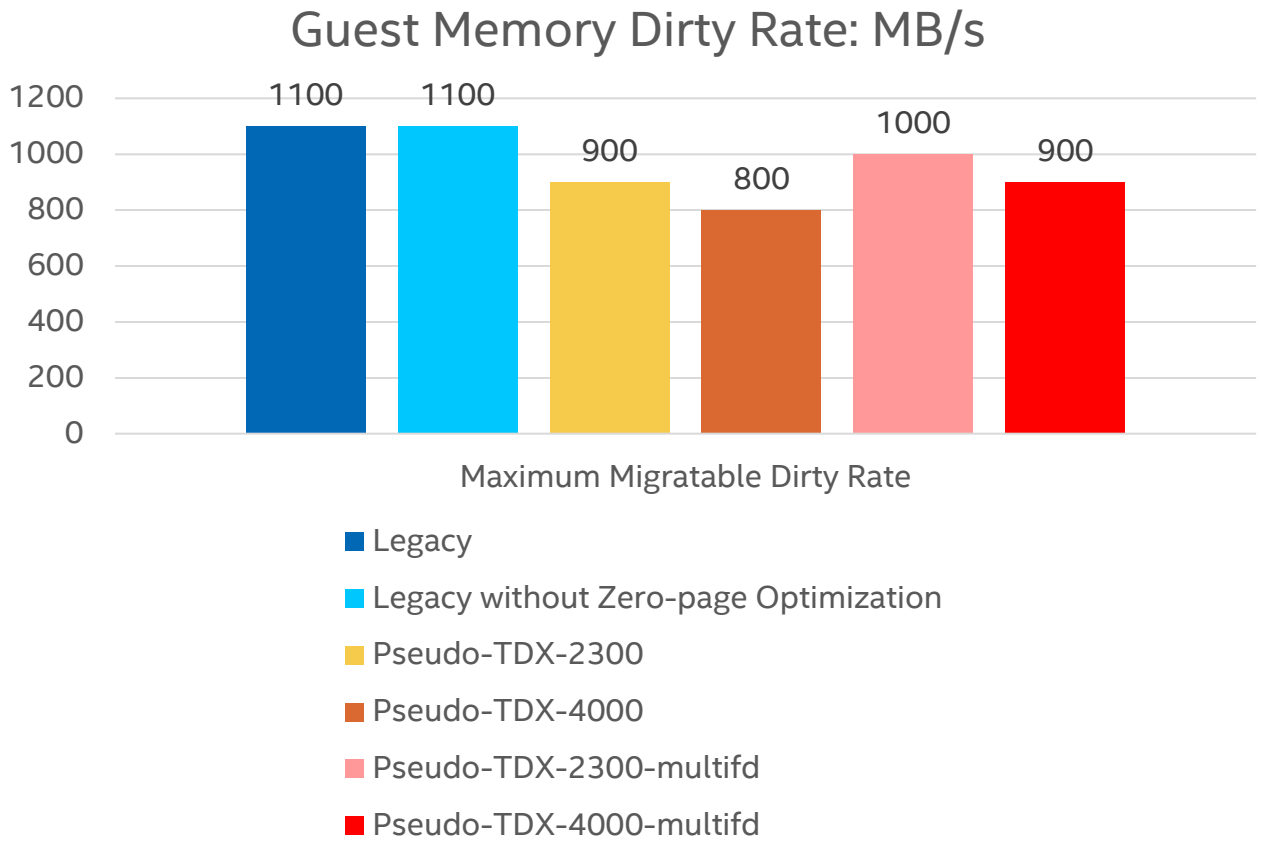
Tests with 600MB/s Memory Dirty Rate

- Running a workload in guest with 600MB/s memory dirty rate
  - Working set is 600MB

	Legacy		Legacy without Zero-page Opt		Pseudo-TDX-2300		Pseudo-TDX-4000		Pseudo-TDX-2300-multifd		Pseudo-TDX-4000-multifd	
Total Migration Time (Seconds)	13.1		30.5		40.6		50.8		34.4		37.1	
Downtime (Milliseconds)	366		355		368		374		372		372	
Dirty Count	5		5		9		20		5		6	
1st Round Migration Throughput (Pages per Second)	733017		282893		214663		176055		267530		251473	
1st Round Network Throughput (Mbps)	52.8		9288.0		7074.8		5780.2		8789.4		8256.8	
CPU Usages (%)	23.5% (vCPUs)	100% (Migration)	23.5% (vCPUs)	75% (Migration)	23.5% (vCPUs)	78.2% (Migration)	23.5% (vCPUs)	78.2% (Migration)	23.5% (vCPUs)	132.2% (Migration)	23.5% (vCPUs)	149.2% (Migration)

Maximum Migratable Dirty Rate

- Guest with memory dirty rate larger than the maximum value fails to be live migrated



Status and Plan

Status and Plan

- Pre-copy enabling
  - Draft code ready, pending to test
  - Plan to post out the patches to the QEMU/KVM mailinglists in Q1'2022
- Multi-fd enabling
  - Create multiple migration streams, which allows multiple iothreads to export/import TD private pages in parallel
  - Plan to start support in Q1'2022
- Post-copy enabling
  - Plan to start support in Q2'2022

Disclaimers

No license (express or implied, by estoppel or otherwise) to any intellectual property rights is granted by this document.

Intel disclaims all express and implied warranties, including without limitation, the implied warranties of merchantability, fitness for a particular purpose, and non-infringement, as well as any warranty arising from course of performance, course of dealing, or usage in trade.

This document contains information on products, services and/or processes in development. All information provided here is subject to change without notice. Contact your Intel representative to obtain the latest forecast, schedule, specifications and roadmaps.

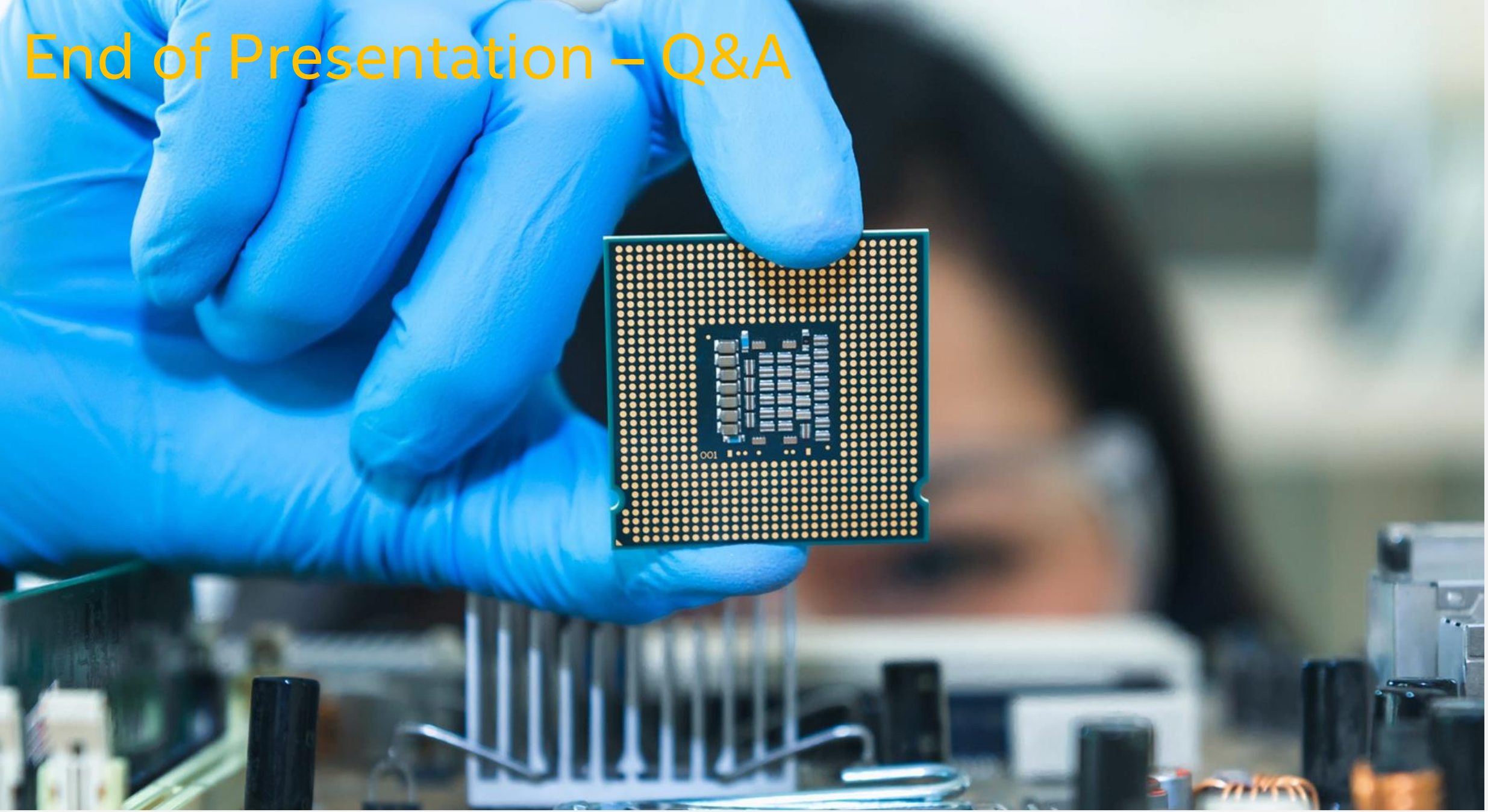
The products and services described may contain defects or errors known as errata which may cause deviations from published specifications. Current characterized errata are available on request. No product or component can be absolutely secure.

Copies of documents which have an order number and are referenced in this document may be obtained by calling 1-800-548-4725 or by visiting www.intel.com/design/literature.htm

Intel, the Intel logo, and Xeon are trademarks of Intel Corporation or its subsidiaries in the U.S. and/or other countries.

*Other names and brands may be claimed as the property of others

End of Presentation – Q&A



Backup

Test 2: Network Throttling – MAX ~3Gbps

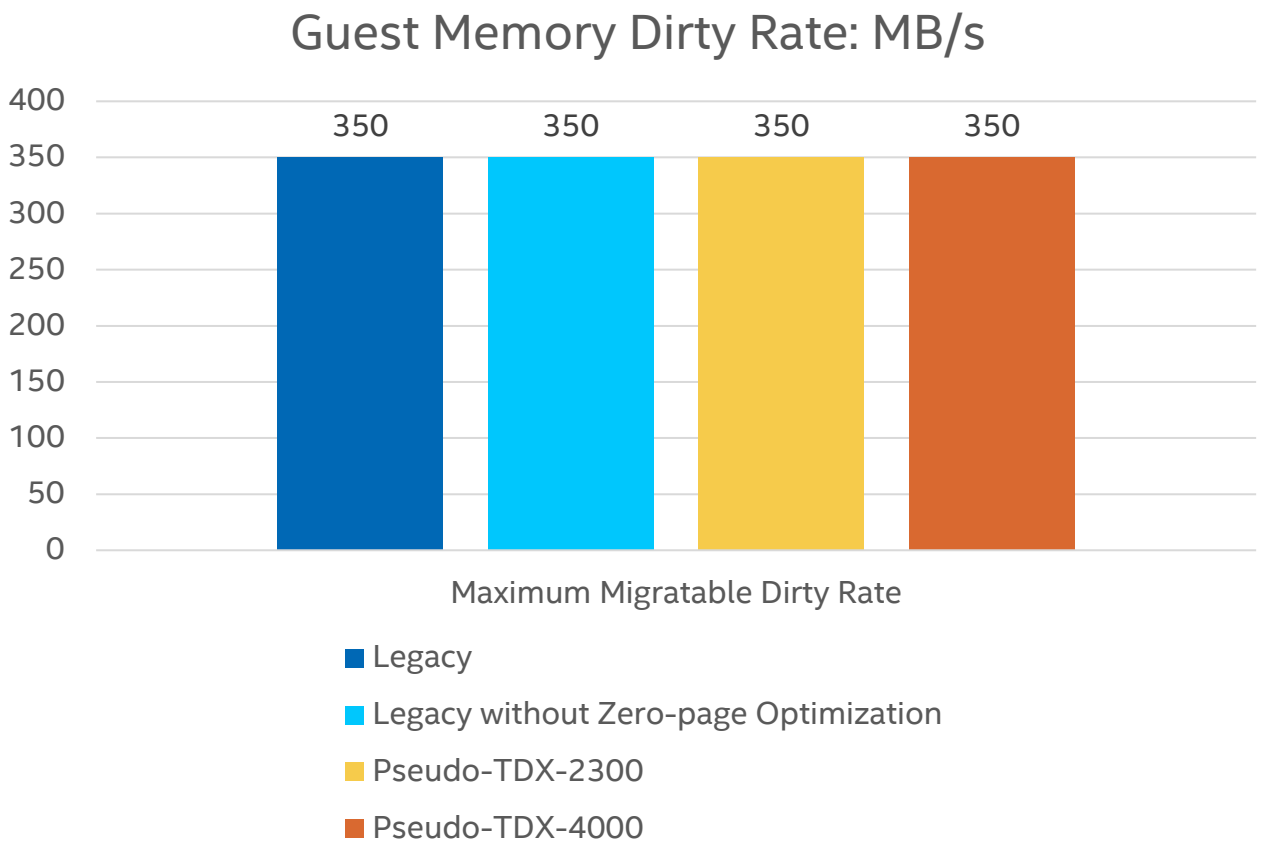
Tests with 300MB/s Memory Dirty Rate

- Running a workload in guest with 300MB/s memory dirty rate
  - Working set is 300MB

	Legacy		Legacy without Zero-page Opt		Pseudo-TDX-2300		Pseudo-TDX-4000	
Total Migration Time (Seconds)	15.6		87.3		87.4		87.4	
Downtime (Milliseconds)	187		177		187		225	
Dirty Count	10		10		10		10	
1st Round Migration Throughput (Pages per Second)	723540		98120		98120		98120	
1st Round Network Throughput (Mbps)	52.1		3221.48		3221.48		3221.48	
CPU Usages (%)	9.6% (vCPUs)	100% (migration)	9.6% (vCPUs)	26.8% (migration)	9.6% (vCPUs)	38.5% (migration)	9.6% (vCPUs)	44.7% (migration)

Maximum Migratable Dirty Rate

- Guest with memory dirty rate larger than the maximum value fails to be live migrated




intel