Persistent Memory Today

Presented by Andy Rudoff, Intel Corporation

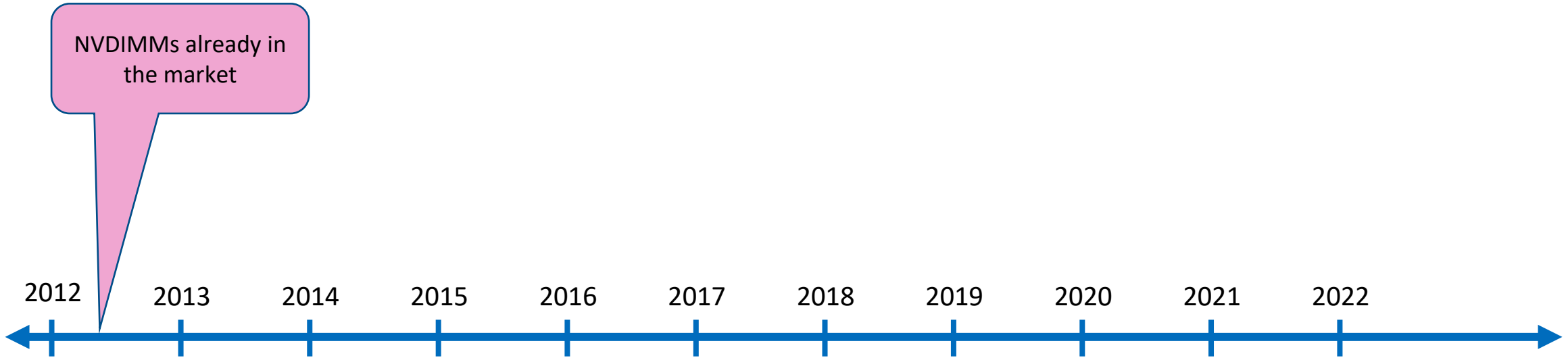
In this talk=

- The story so far
- A surprising focus for PMem
- How the SNIA PMem Programming Model turned out
- Future directions for PMem

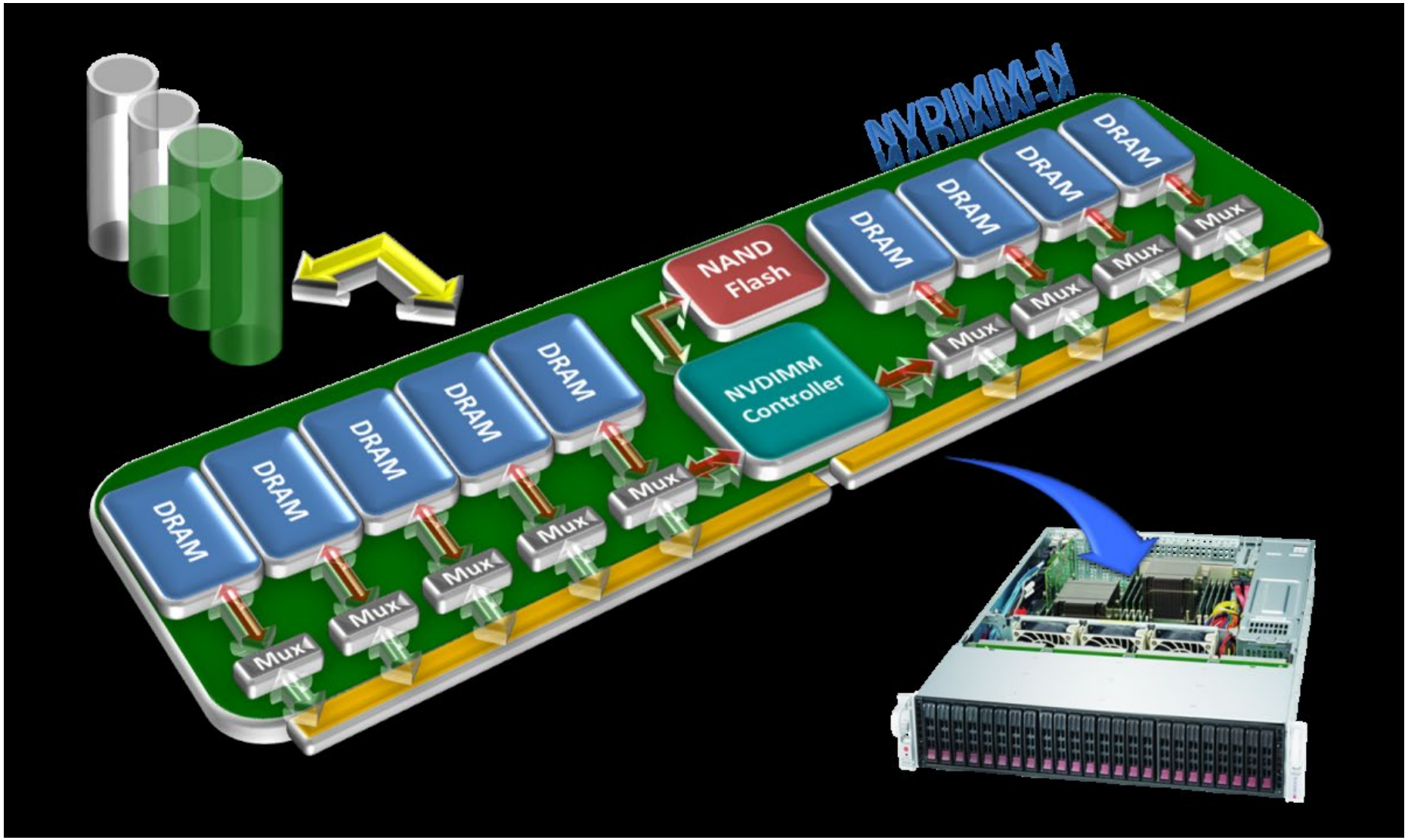
In this talk…

- The story so far
- A surprising focus for PMem
- How the SNIA PMem Programming Model turned out
- Future directions for PMem

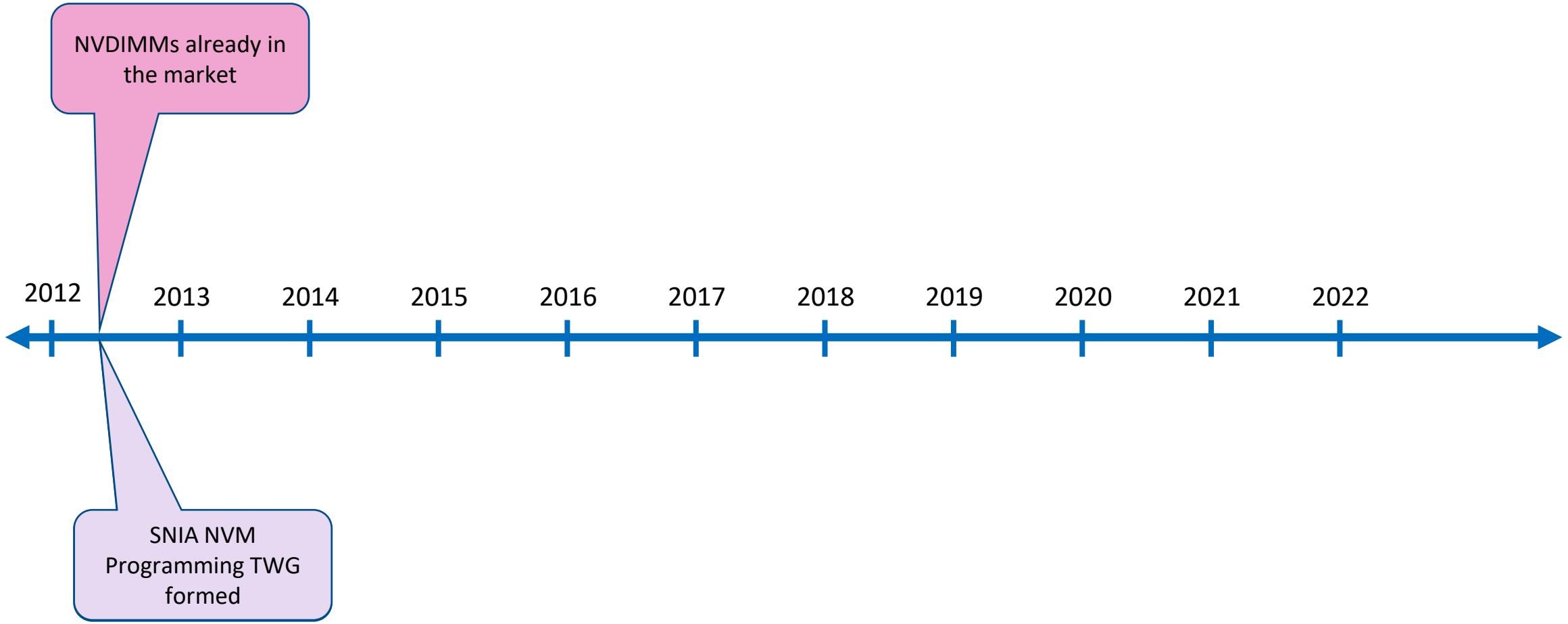
The Timeline



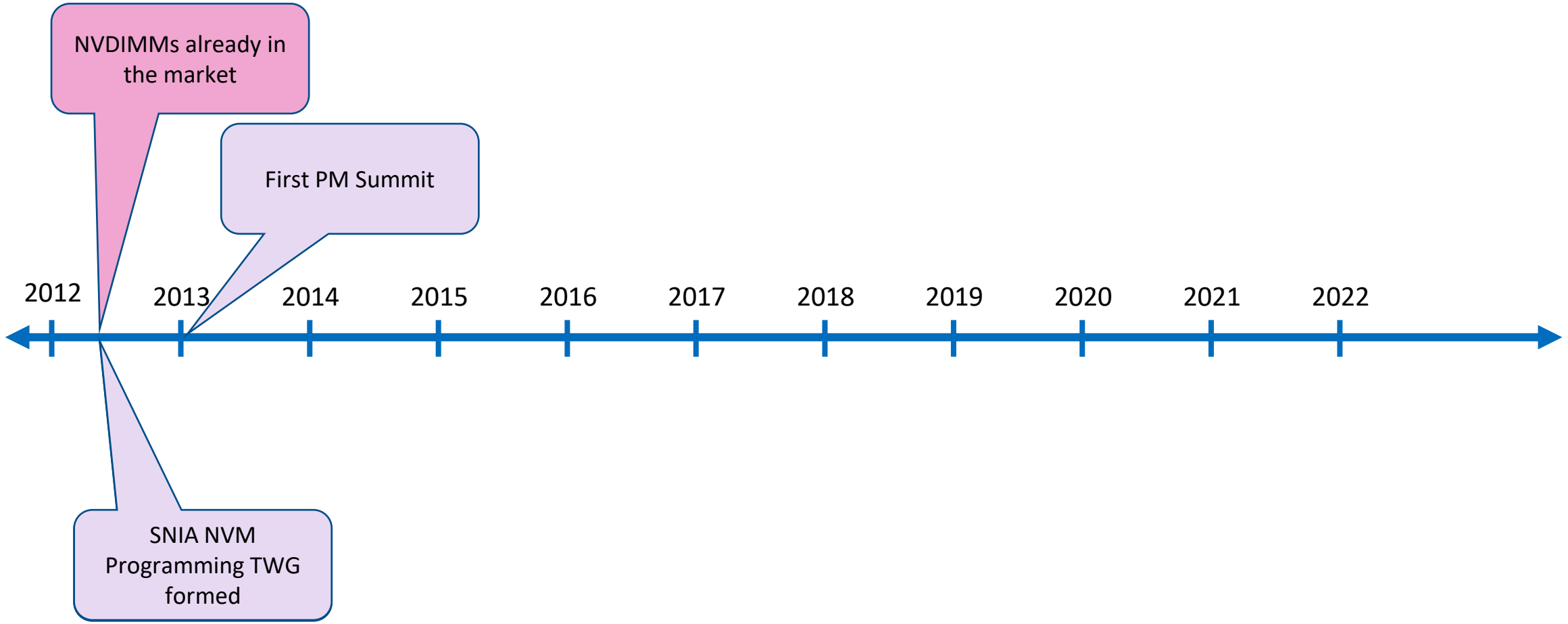
NVDIMM-N



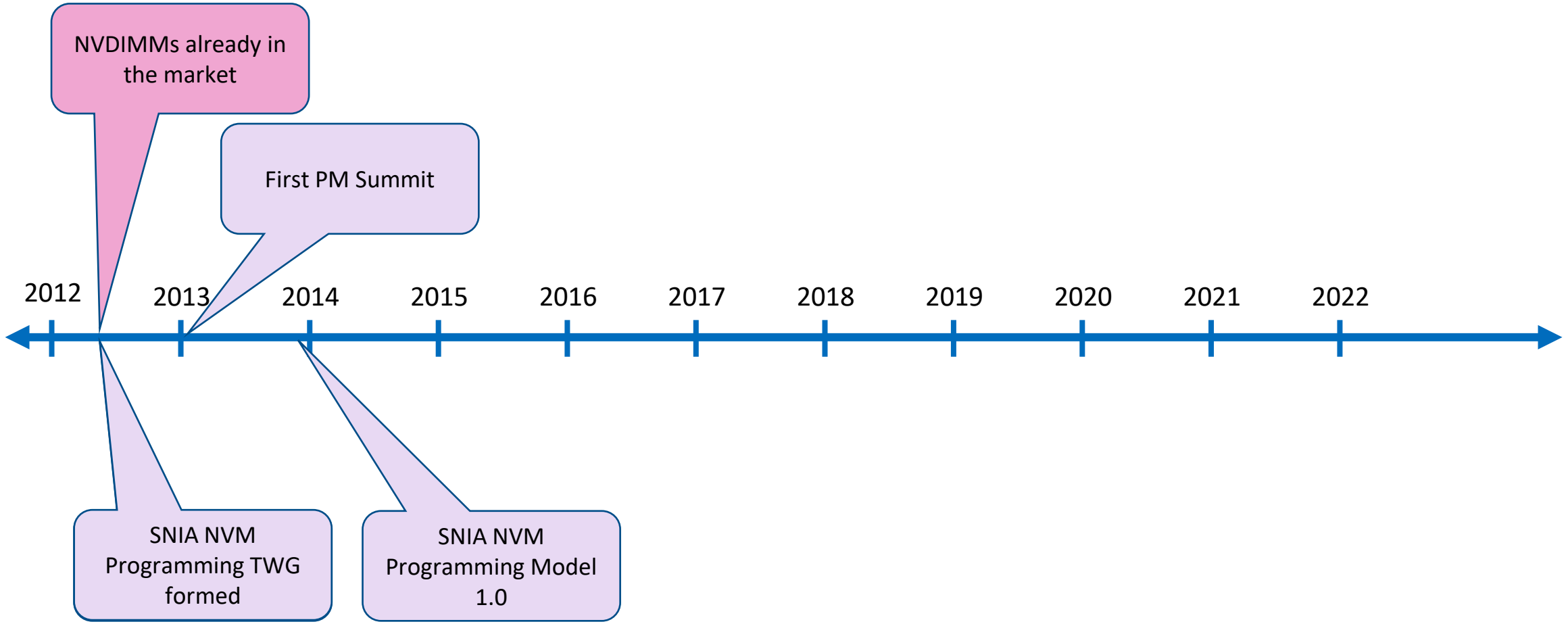
The Timeline



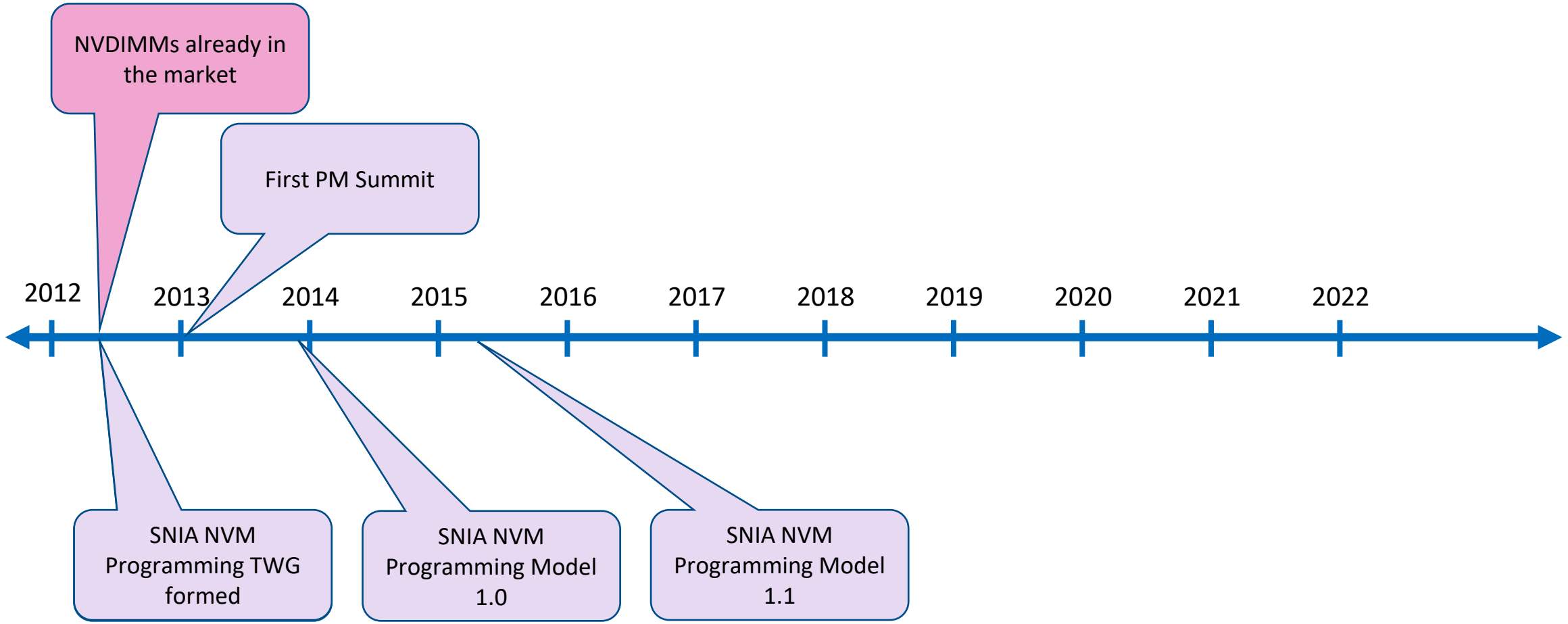
The Timeline



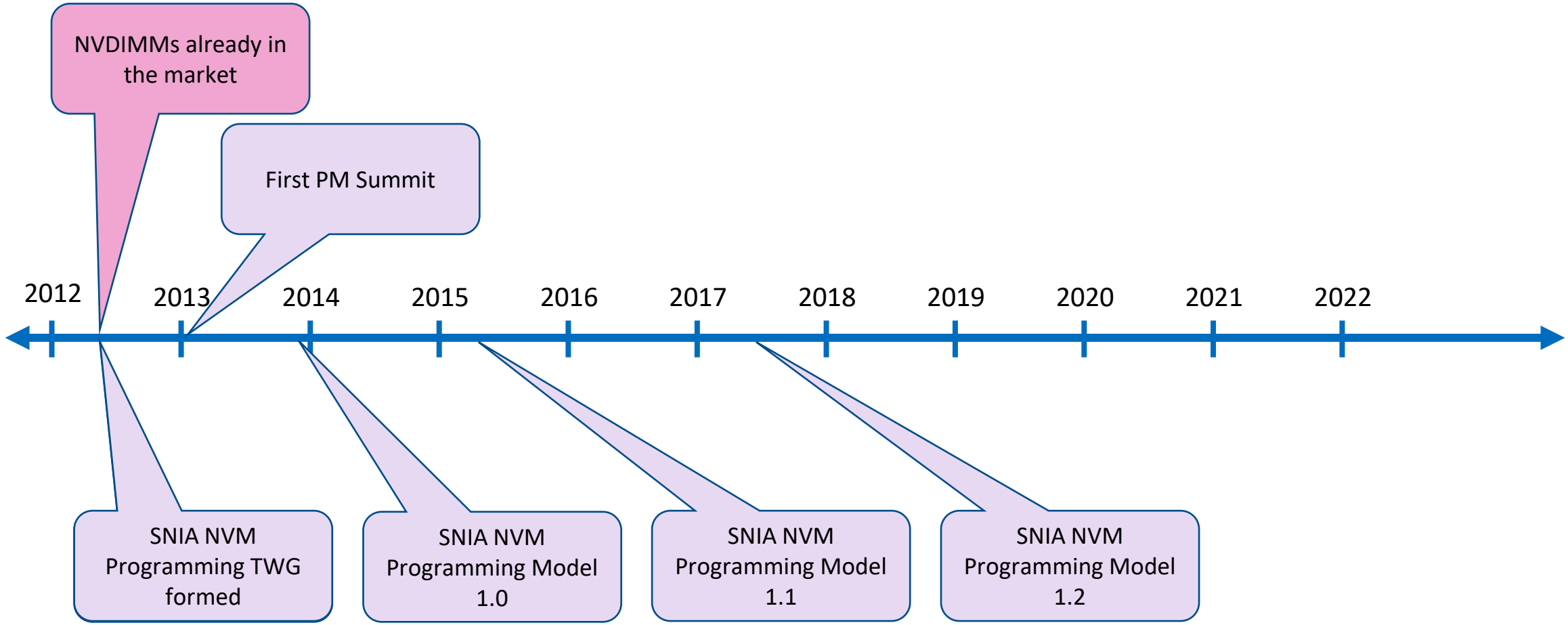
The Timeline



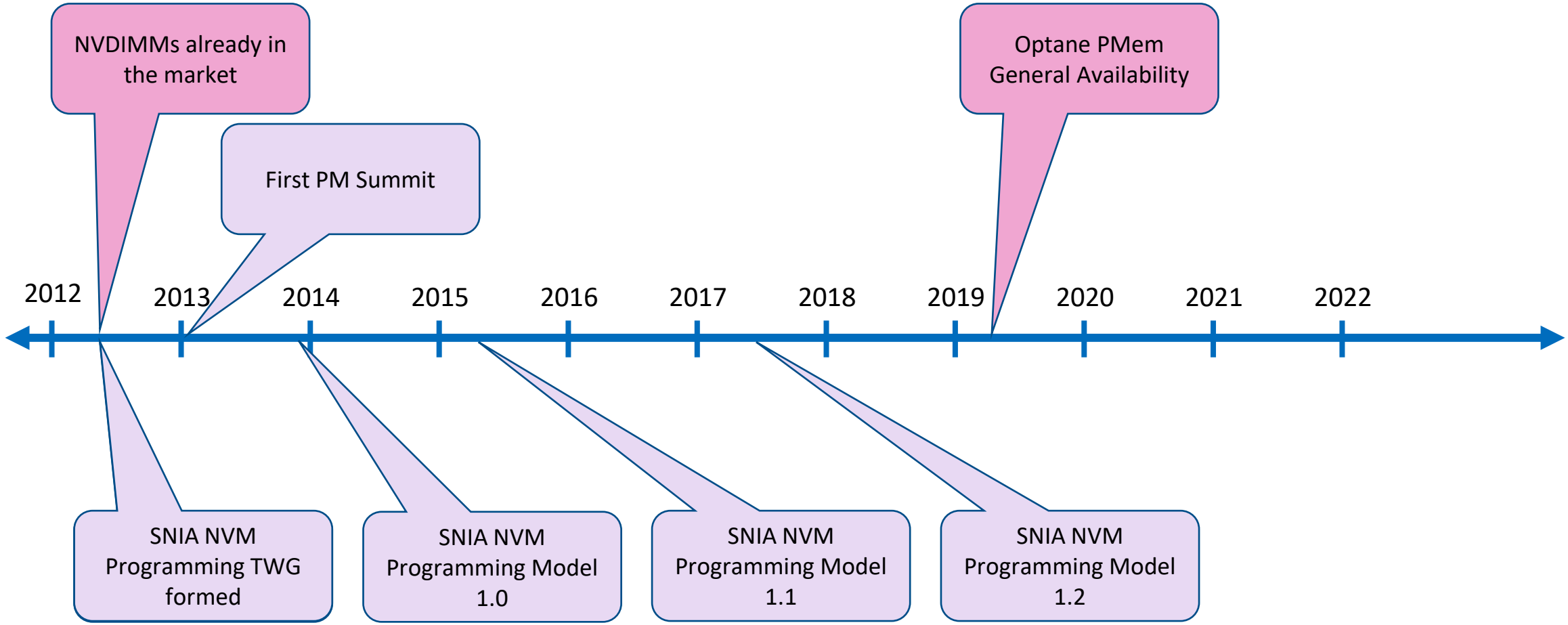
The Timeline



The Timeline

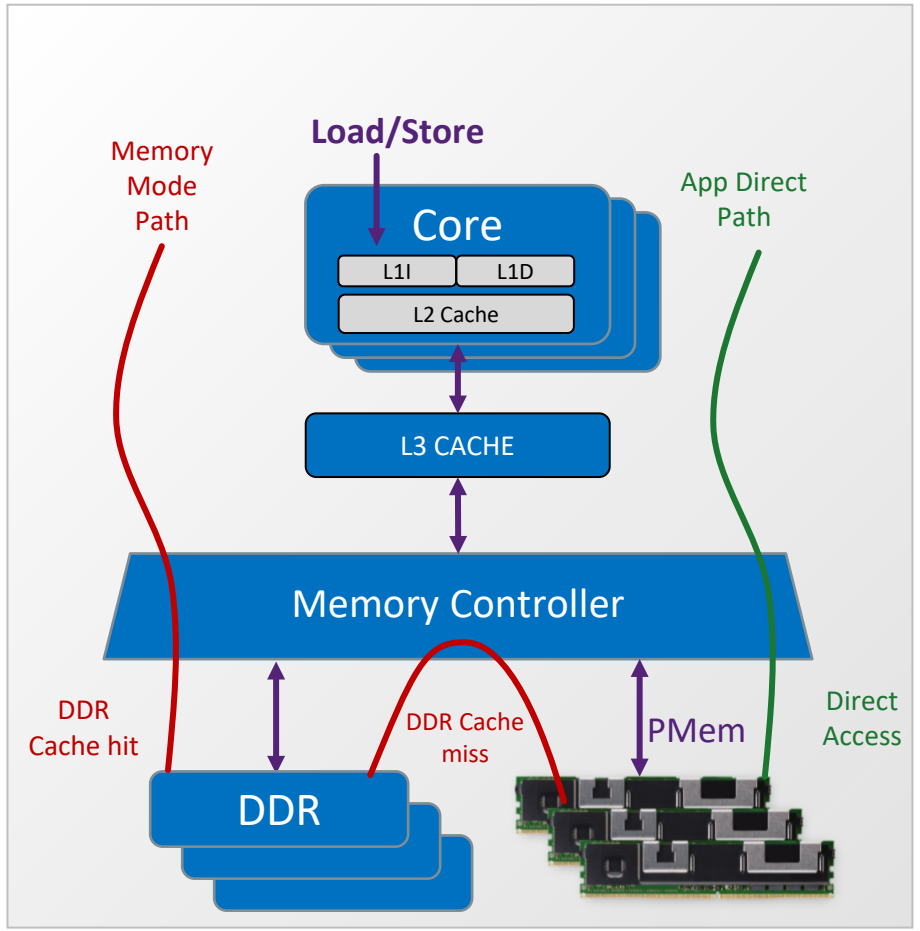


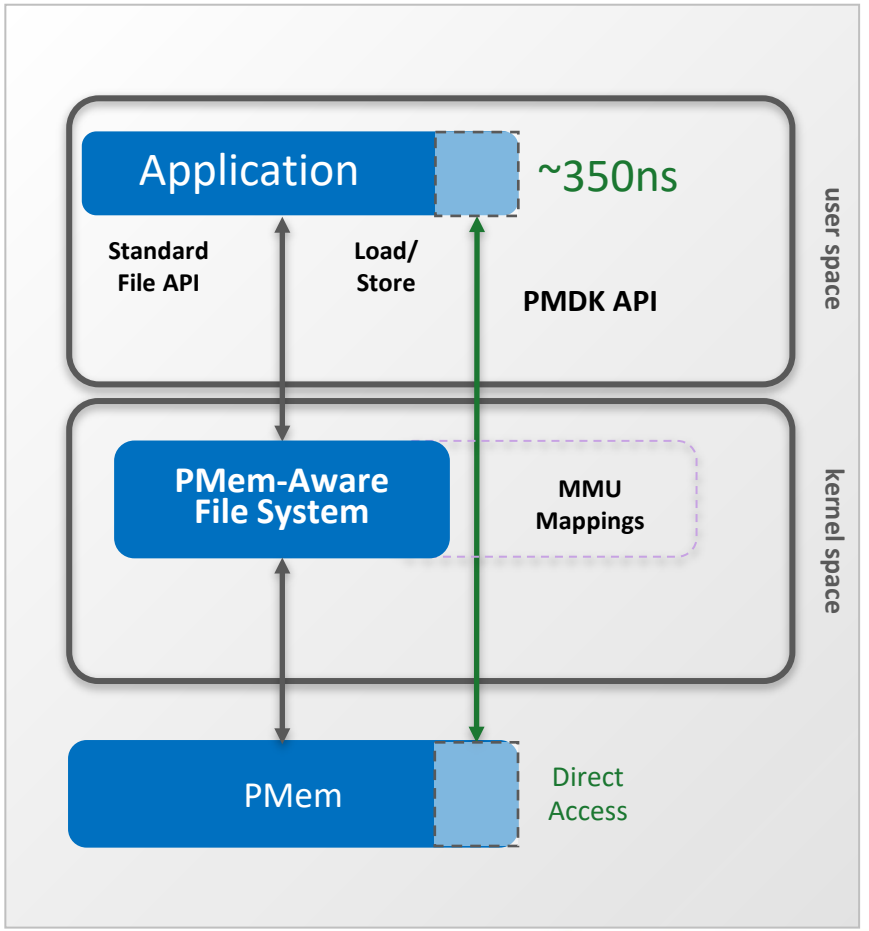
The Timeline



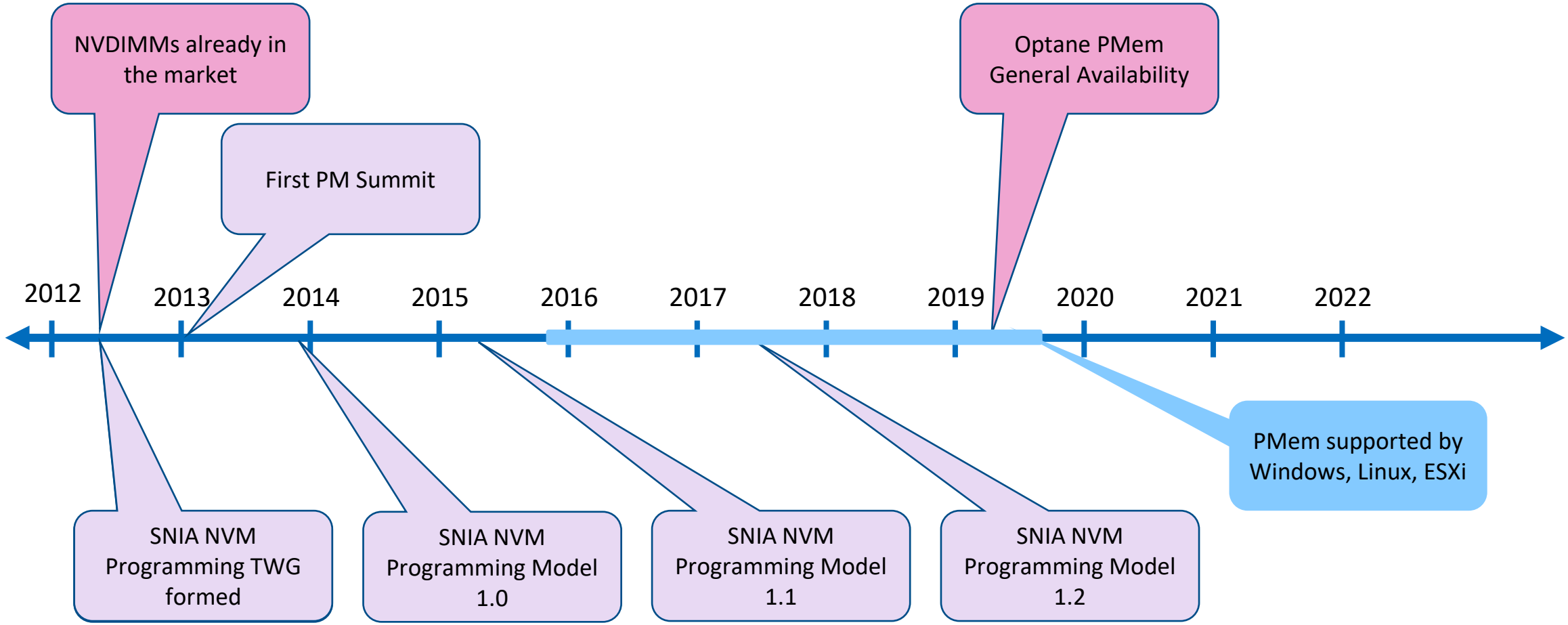
Connecting to the Memory Bus

Intel's approach with Optane PMem

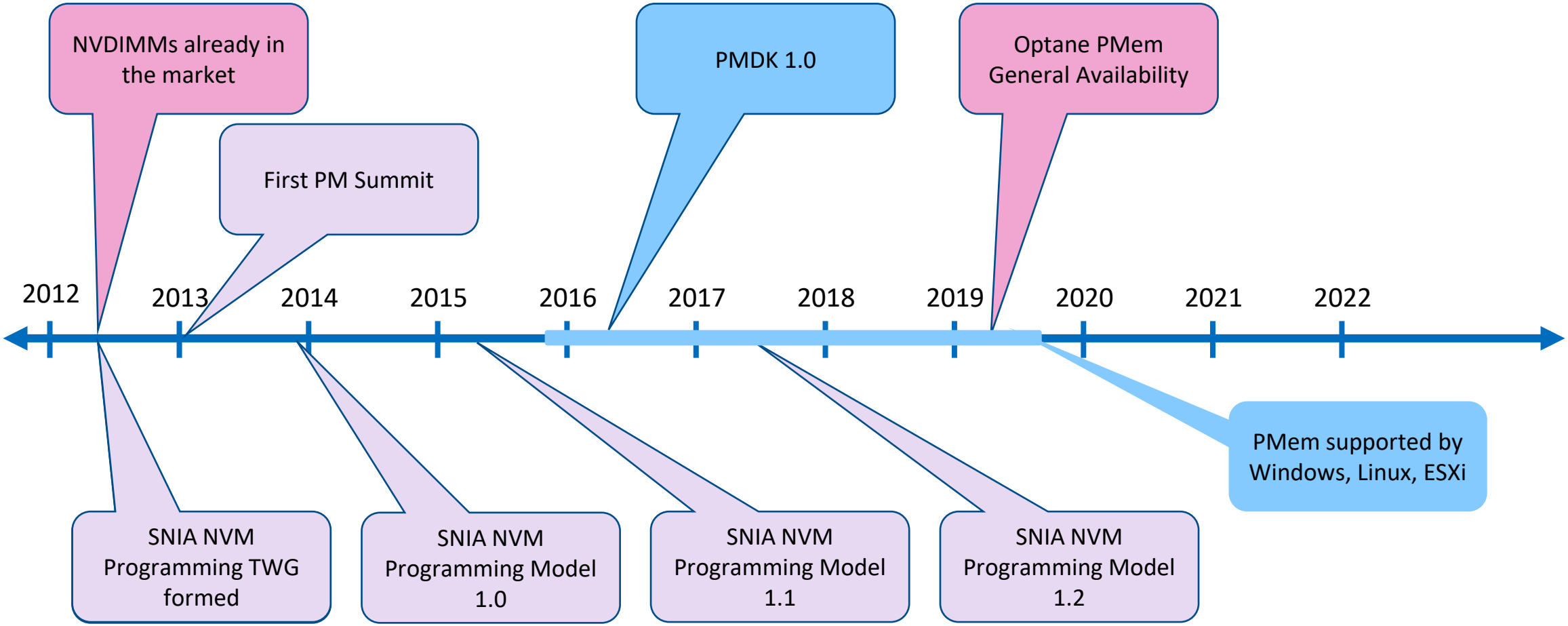




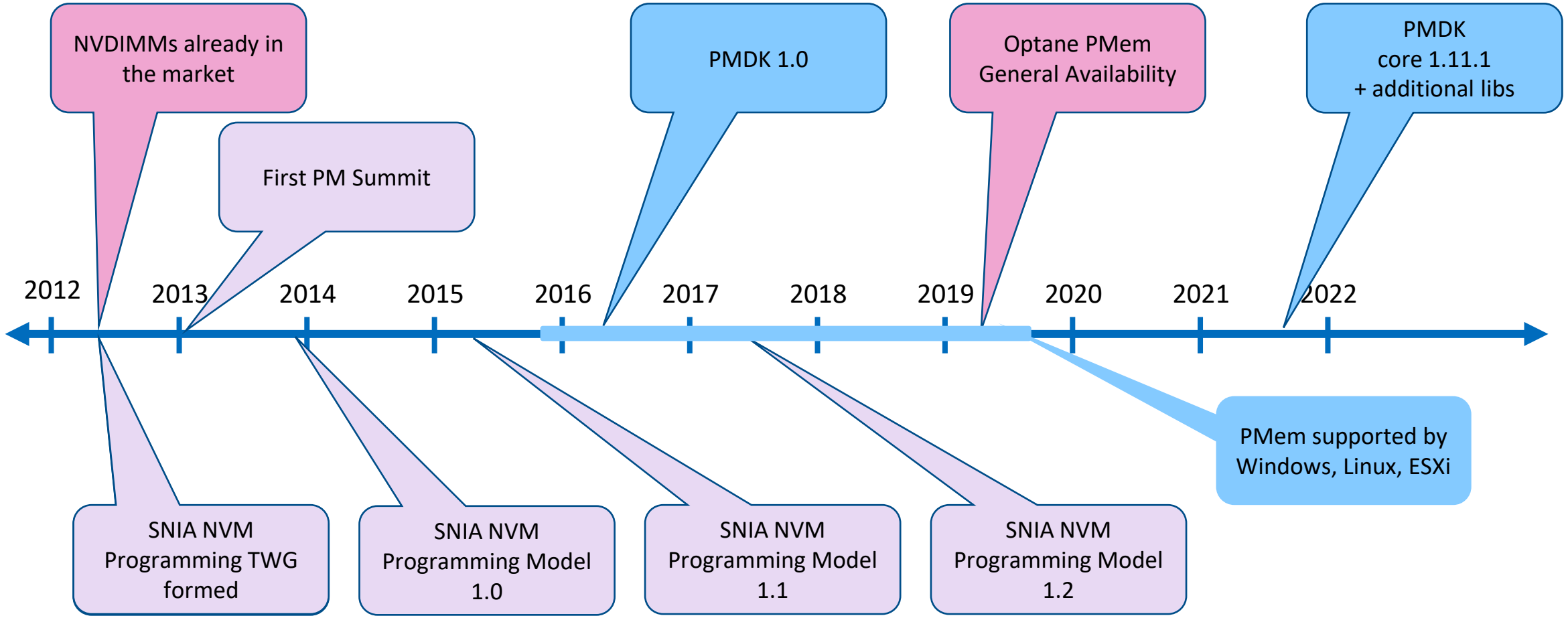
The Timeline



The Timeline



The Timeline



Persistent Memory Development Kit – pmem.io

PCJ – Persistent Collection for Java

PMUL

C++ C LLPL Python

pmemkv

C C++ Java JS Ruby

Transaction Support

Interface to create a persistent memory resident log file

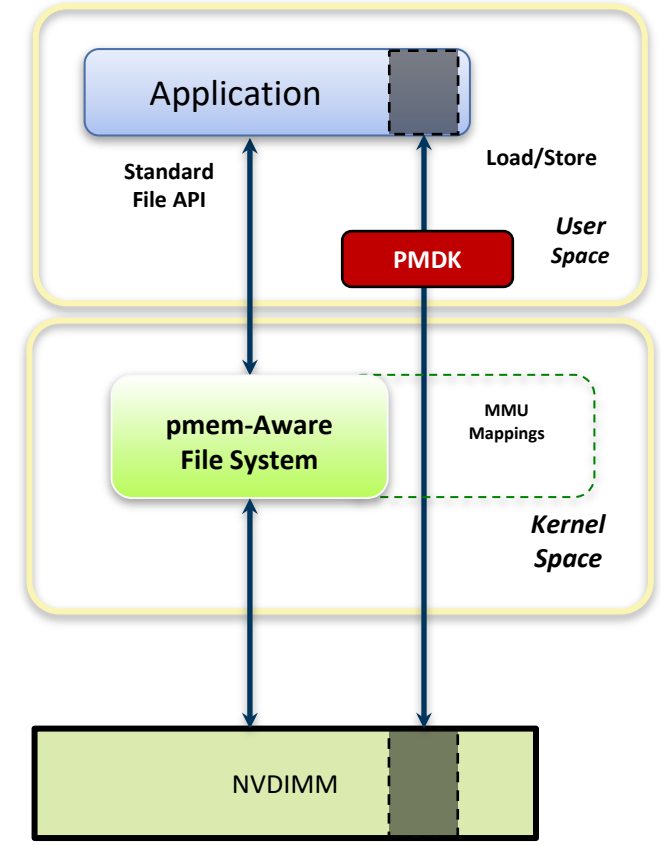
libpmemlog

Interface for persistent memory allocation, transactions and general facilities

libpmemobj

Interface to create arrays of pmem-resident blocks, of same size, atomically updated

libpmemblk



Support for **volatile** memory usage

memkind

vmemcache

Low level support for local persistent memory

libpmem

libpmem2

Low level support for remote persistent memory

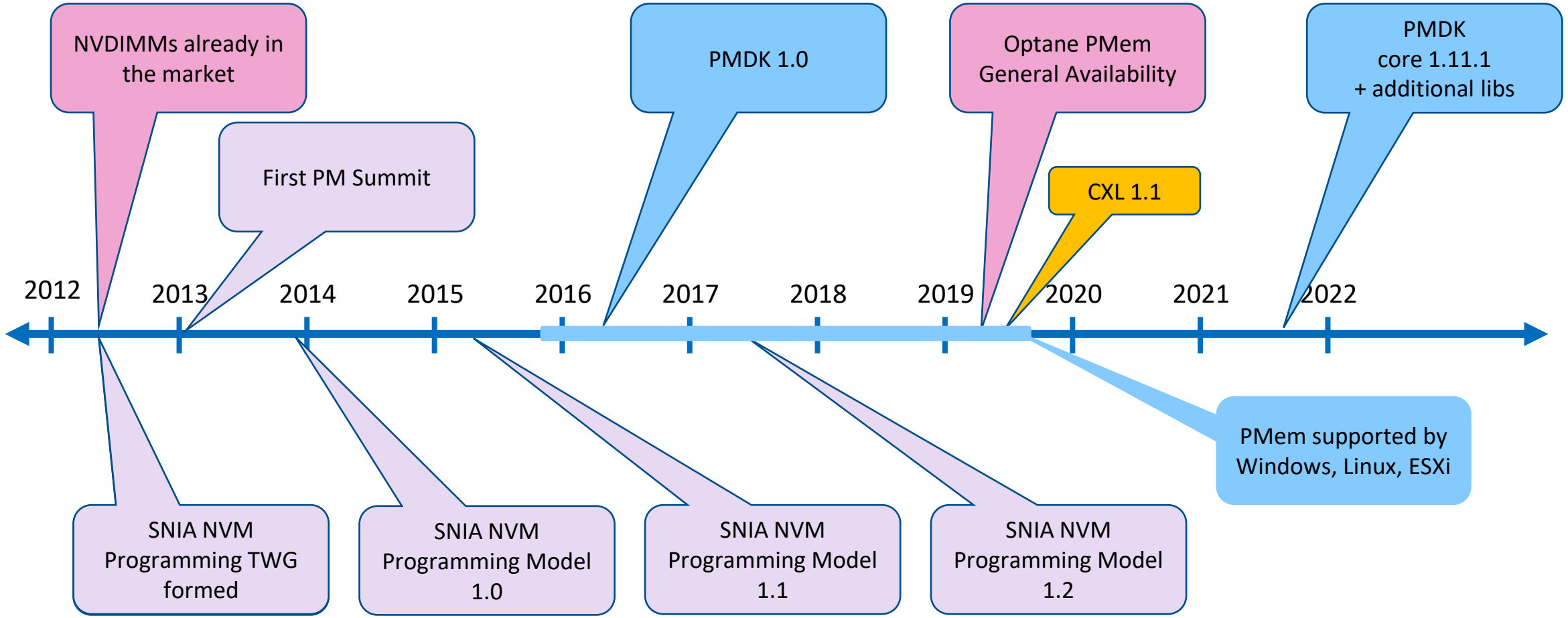
librpmem

librpma

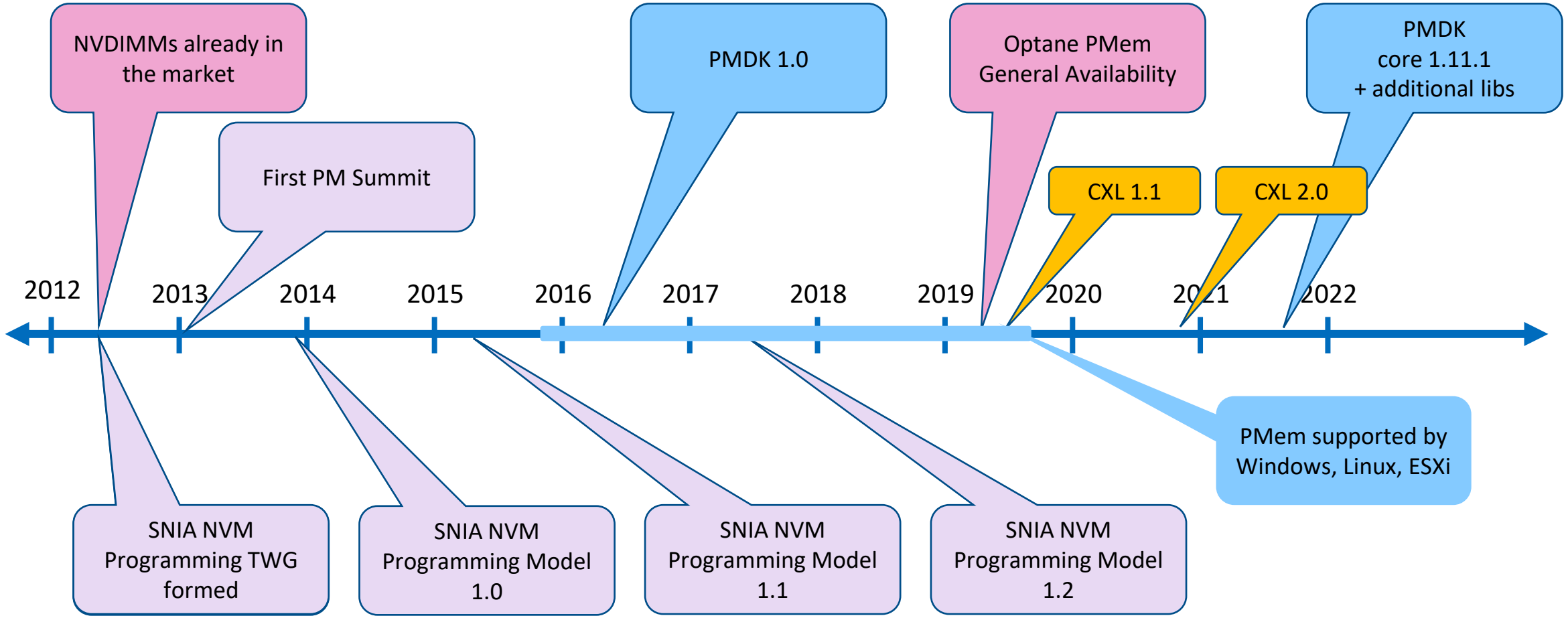
Low-level support

In Development

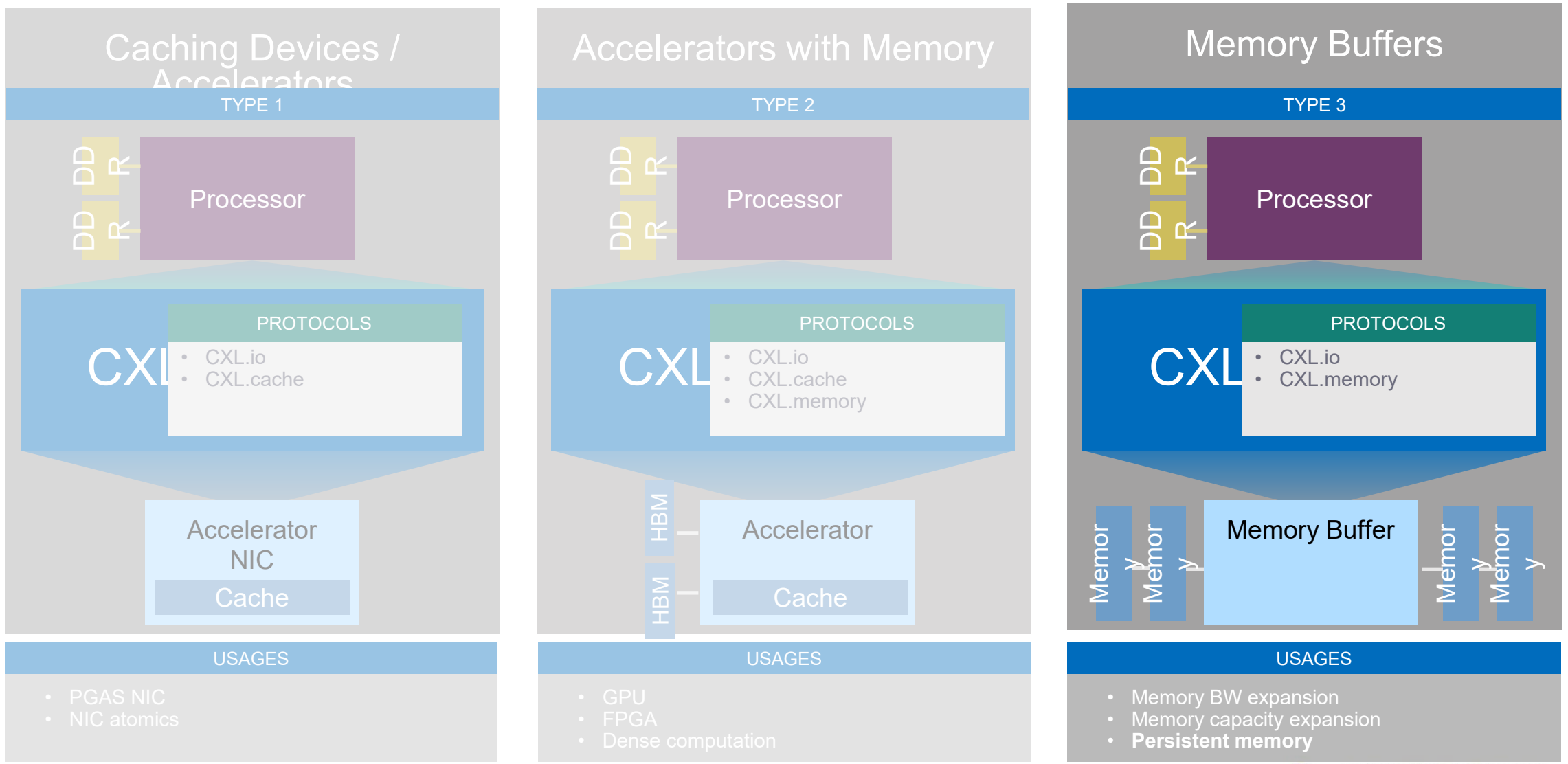
The Timeline



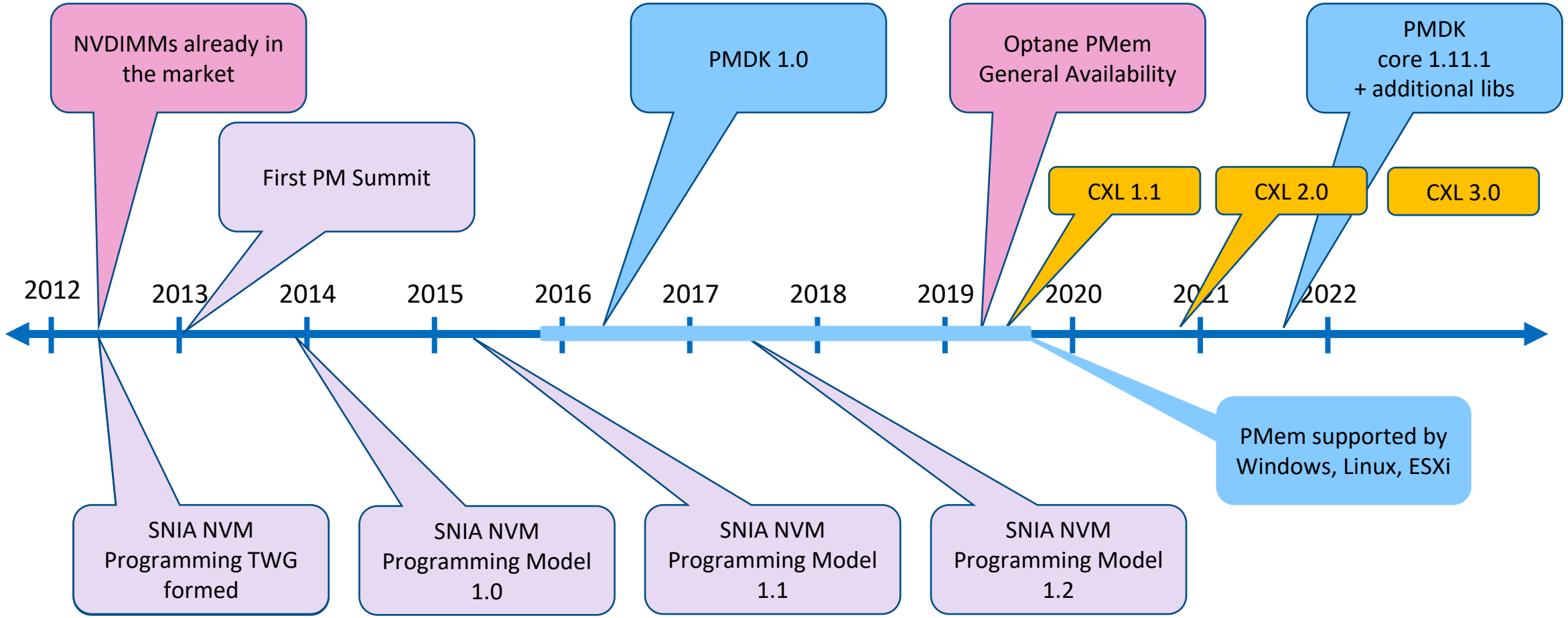
The Timeline



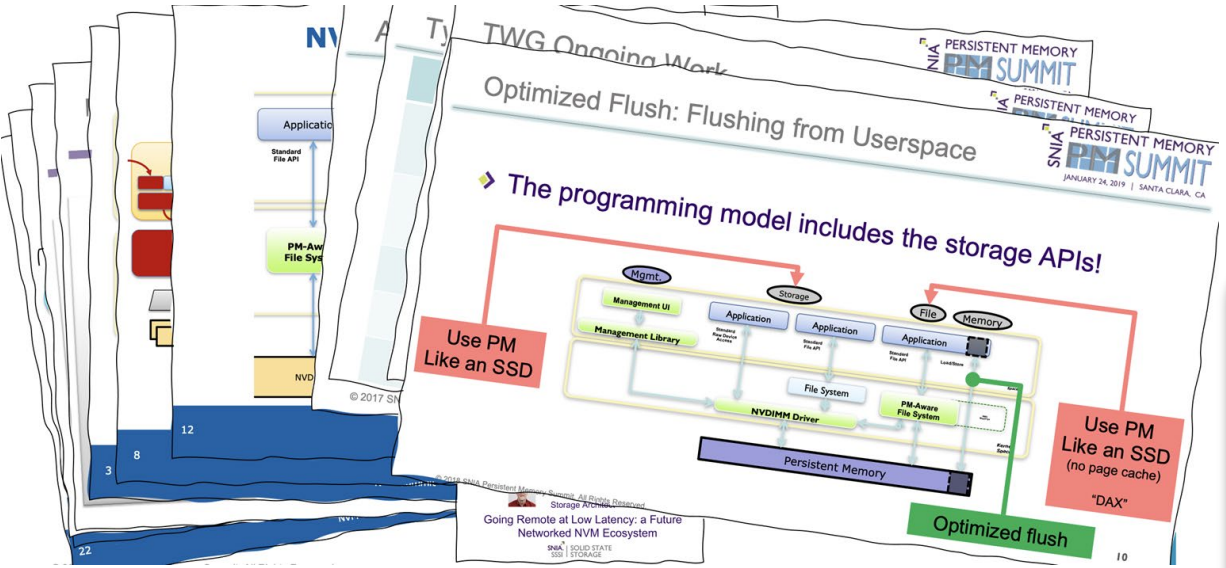
PMem on CXL

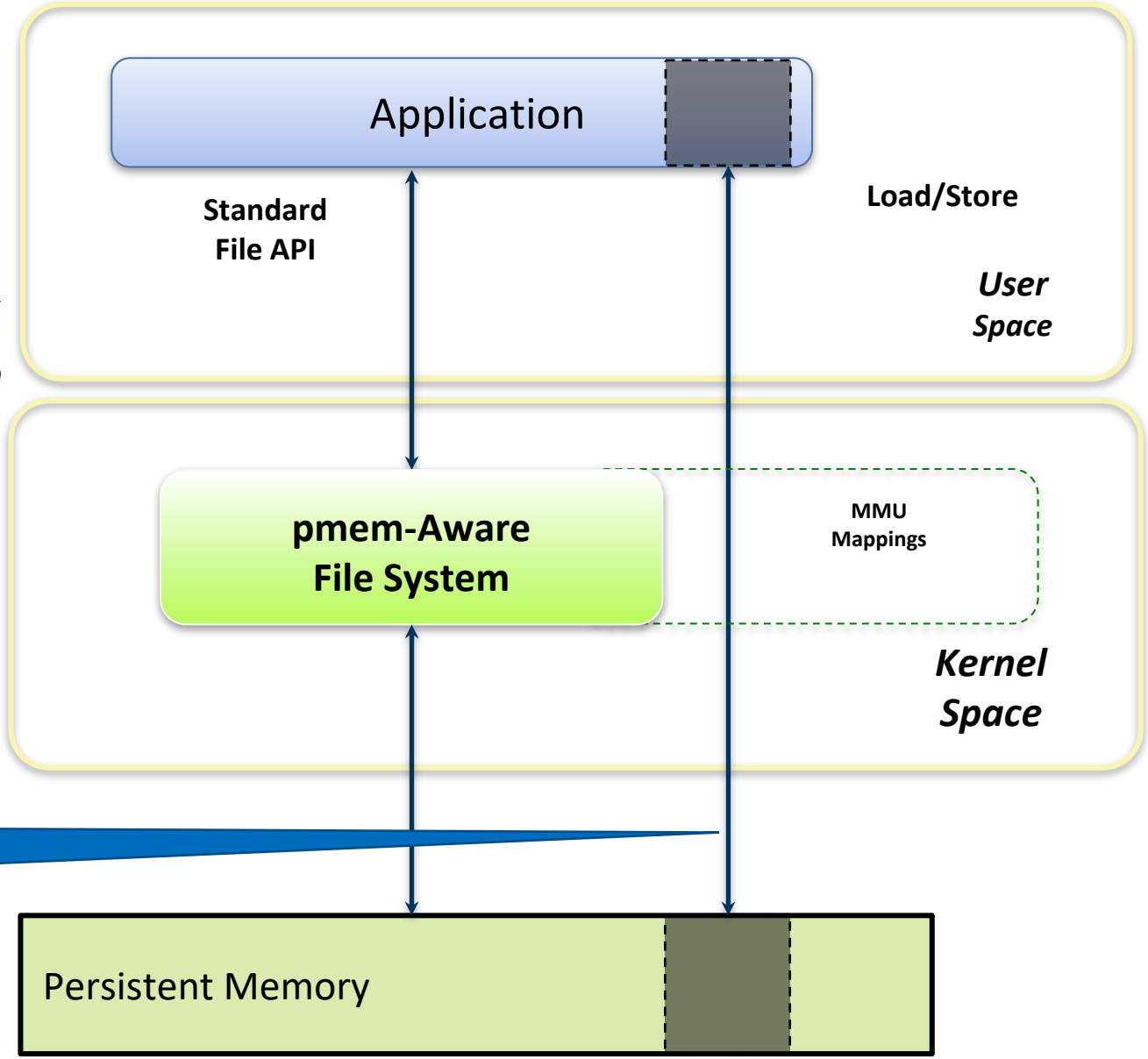


The Timeline



DAX





In this talk…

- The story so far
- A surprising focus for PMem
- How the SNIA PMem Programming Model turned out
- Future directions for PMem

PMem Use Cases

Persistent	Volatile	
I [Fully Exploit PMem Capabilities] Oracle Exadata SAP Hana DAOS	II [App Direct Volatile] Redis/MemKeyDB	Non-Transparent (Apps modified to use pmem)
III [Storage over App Direct] Low-latency storage	IV [Memory Mode] [Kernel Memory Tiering] Big Memory Applications Increased Guest VM capacity	Transparent (No app modifications)

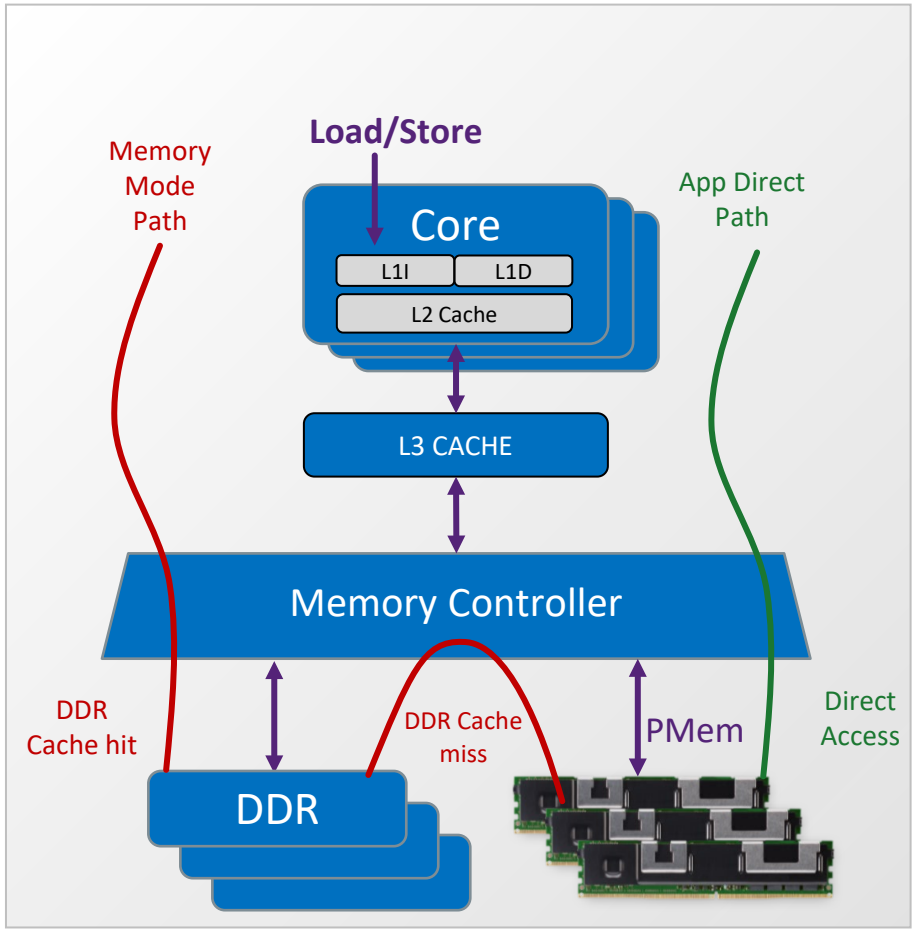
PMem Use Cases

Memory Tiering

Persistent	Volatile	
I [Fully Exploit PMem Capabilities] Oracle Exadata SAP Hana DAOS	II [App Direct Volatile] Redis/MemKeyDB	Non-Transparent (Apps modified to use pmem)
III [Storage over App Direct] Low-latency storage	IV [Memory Mode] [Kernel Memory Tiering] Big Memory Applications Increased Guest VM capacity	Transparent (No app modifications)

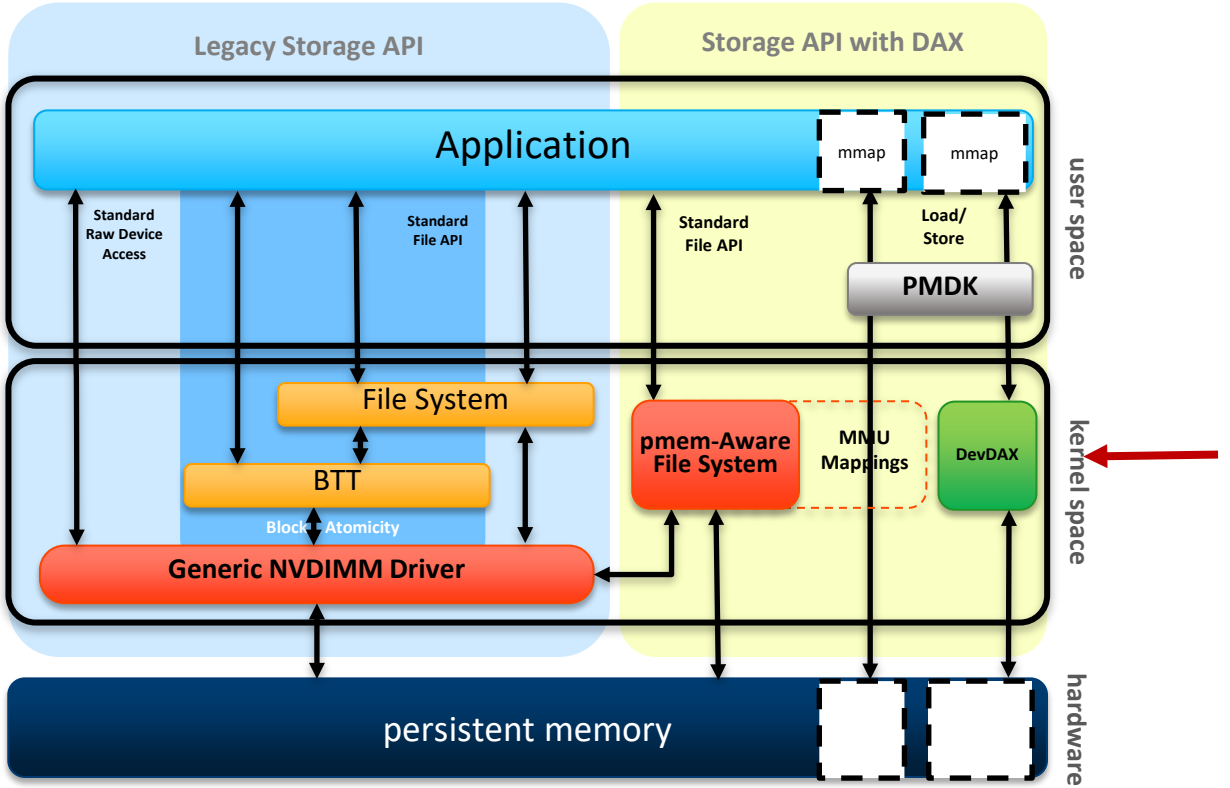
Transparent Memory Tiering – an incomplete list

- Memory Mode
  - Transparent even to OS (mostly)



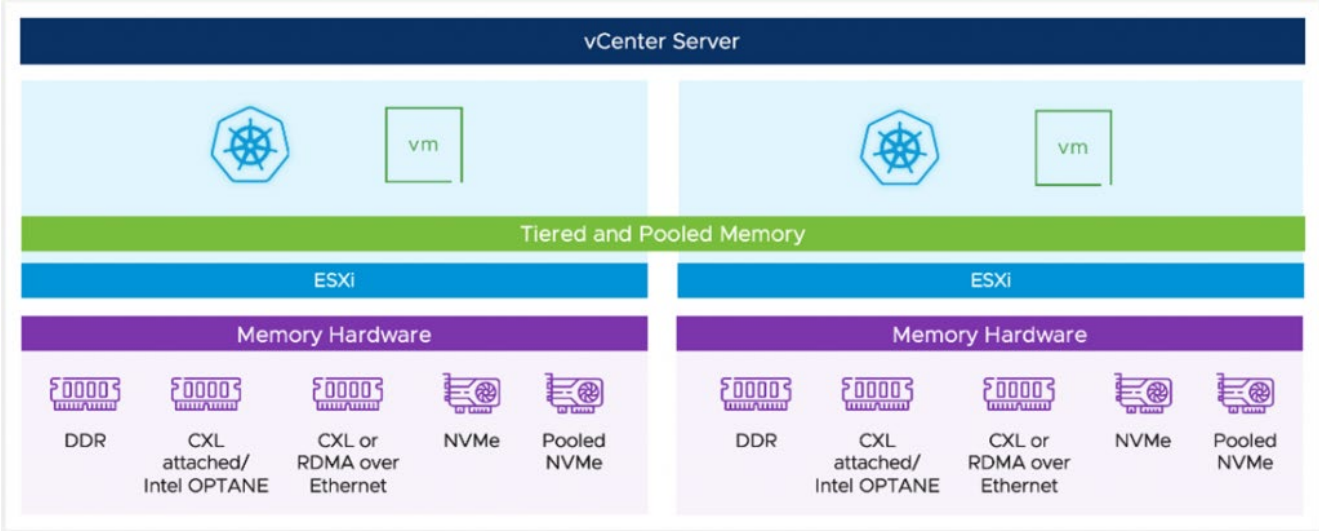
Transparent Memory Tiering – an incomplete list

- Memory Mode
  - Transparent even to OS (mostly)
- Linux Kernel Memory Tiering
  - Recently upstreamed (mostly)
  - For PMem, leverages Device DAX



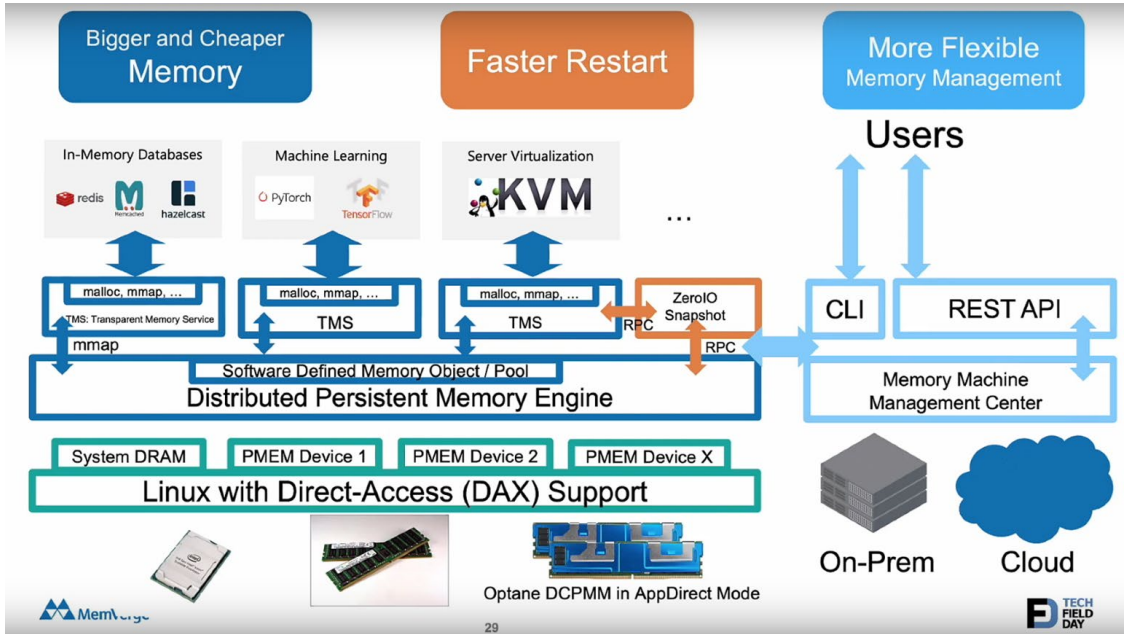
Transparent Memory Tiering – an incomplete list

- Memory Mode
  - Transparent even to OS (mostly)
- Linux Kernel Memory Tiering
  - Recently upstreamed (mostly)
  - For PMem, leverages Device DAX
- VMware Project Capitola
  - Technology preview
  - Aggregate tiers of different memory types



Transparent Memory Tiering – an incomplete list

- Memory Mode
  - Transparent even to OS (mostly)
- Linux Kernel Memory Tiering
  - Recently upstreamed (mostly)
  - For PMem, leverages Device DAX
- VMware Project Capitola
  - Technology preview
  - Aggregate tiers of different memory types
- MemVerge
  - User space solution
  - Also leverages persistence!



See Charles Fan:
Persistent Memory Breaks Though the Clouds
Tomorrow

Non-transparent Memory Tiering

- Still might be transparent to the “end application”
  - Example: a key-value store implements tiering, users of API unmodified
- Often performs best of all
  - A few informed decisions about data placement make a huge difference
  - Data movement is tricky/expensive without application involvement
- Can combine with transparent solutions
  - Example: Linux Memory Tiering and libnuma
  - Example: The MemVerge API

See Alessandro Goncalves:
Make Sense of Memory Tiering
This Afternoon

In this talk…

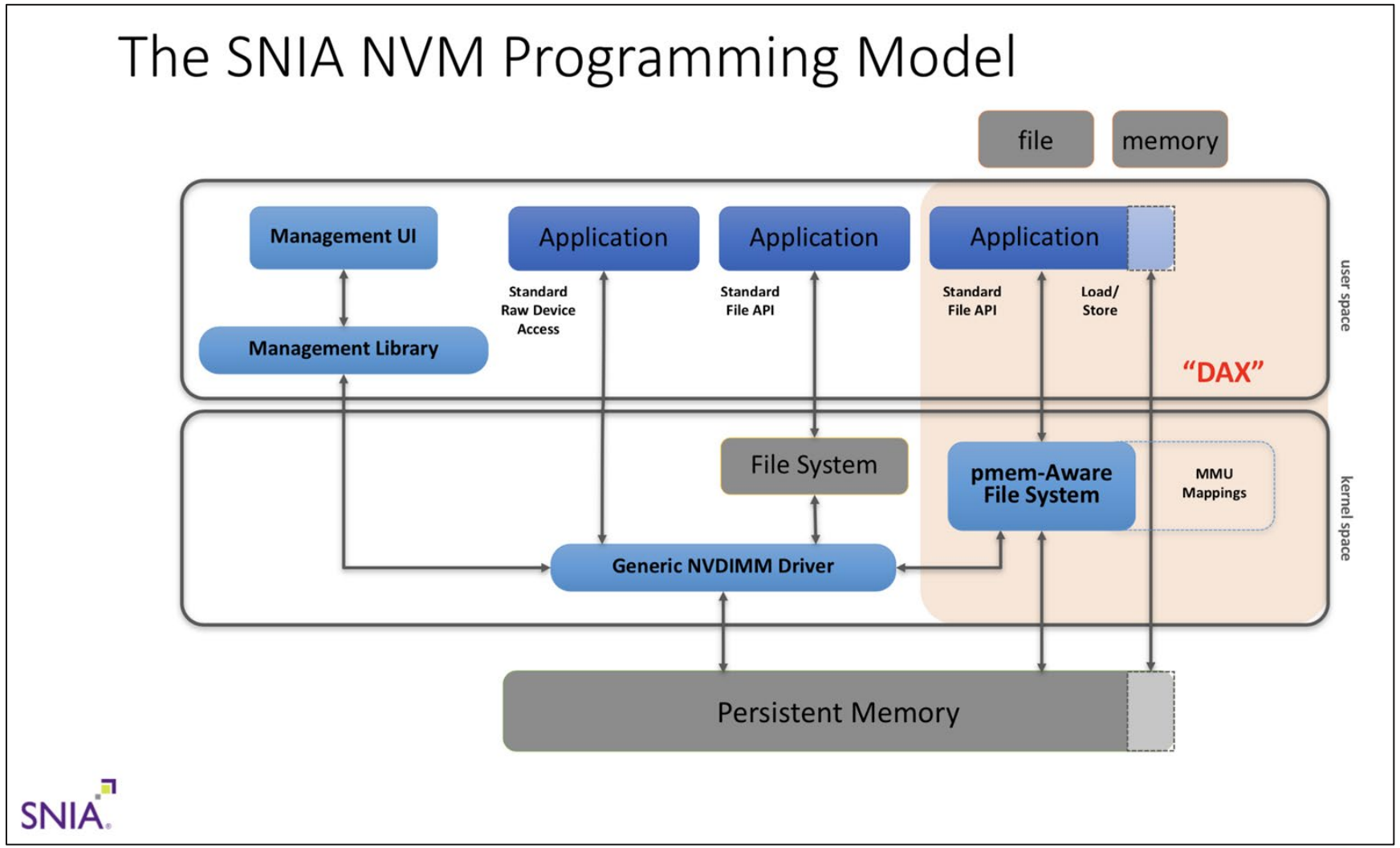
- The story so far
- A surprising focus for PMem
- How the SNIA PMem Programming Model turned out
- Future directions for PMem

PMem Use Cases

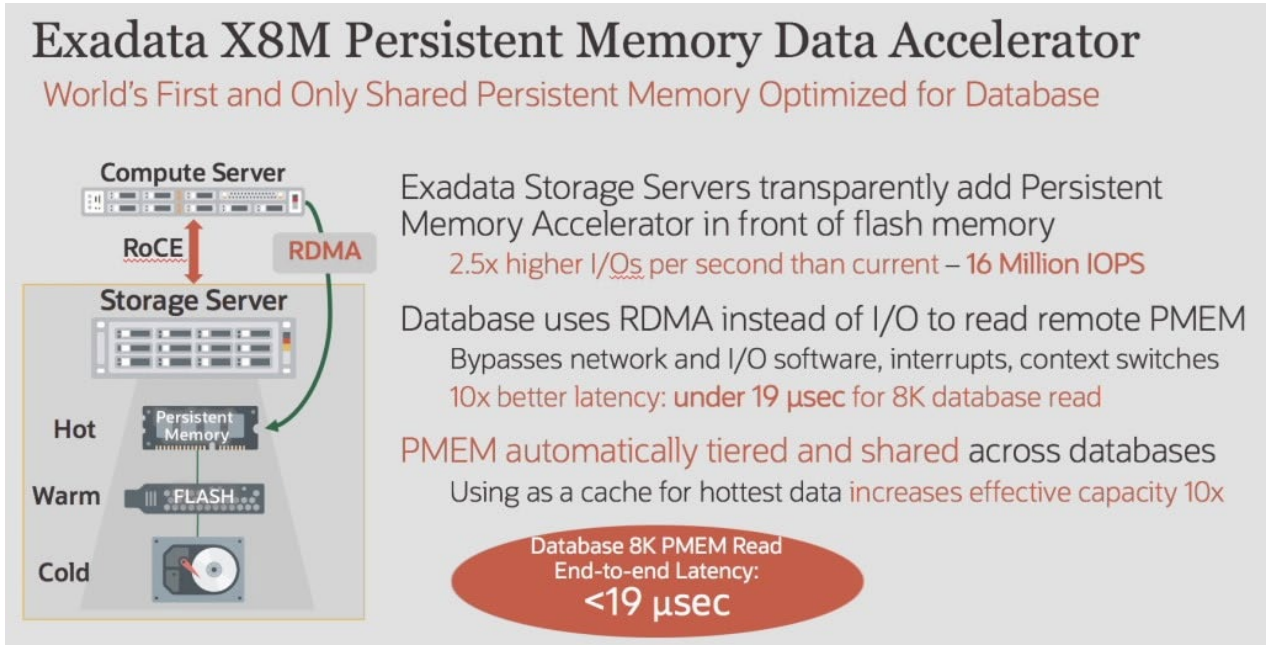
Using the PMem Programming Model

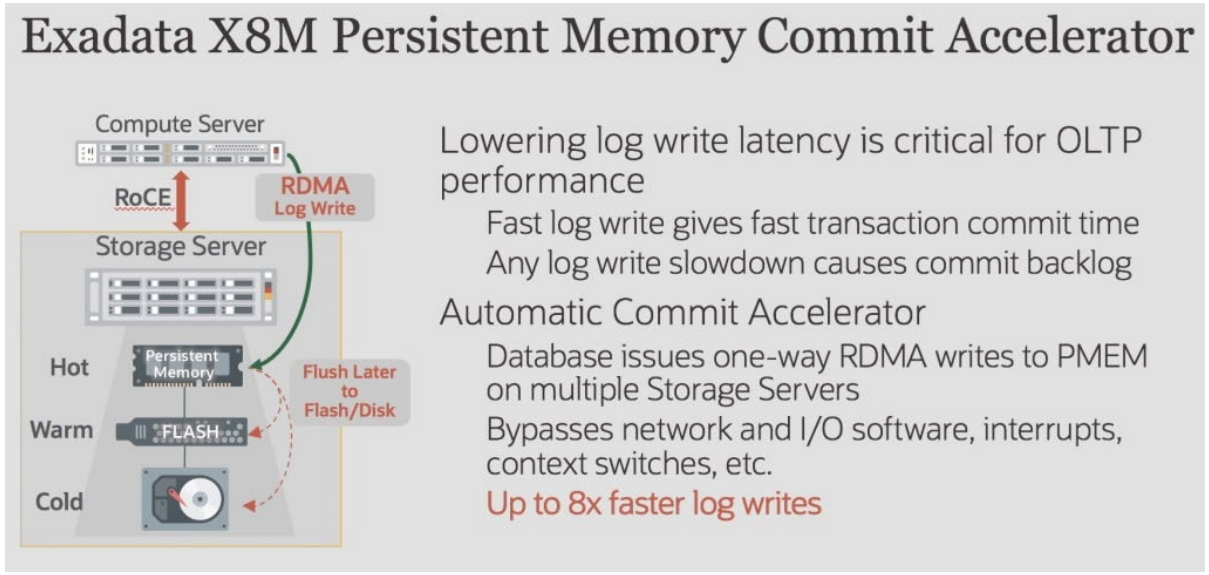
	Persistent	Volatile	
	I [Fully Exploit PMem Capabilities] Oracle Exadata SAP Hana DAOS	II [App Direct Volatile] Redis/MemKeyDB	Non-Transparent (Apps modified to use pmem)
	III [Storage over App Direct] Low-latency storage	IV [Memory Mode] [Kernel Memory Tiering] Big Memory Applications Increased Guest VM capacity	Transparent (No app modifications)

Leveraging Persistence



Oracle Exadata - previously presented at the PM Summit





- Notice the RDMA!
  - DMA works with PMem, so
  - RDMA works with PMem
  - Increasing RDMA activity (librpma)

See VMware's talk:
Accelerating Oracle Workloads…
This Afternoon

Tencent Cloud

RDMA!

Reduction in overall write latency from 120 to 50 microseconds.[1]

Reduction in overall Read latency from 130 to 40 microseconds.[2]

Tencent Cloud Optimizes its Cloud Block Storage to Create an Ultra-fast Experience

As one of the world's leading cloud service providers, Tencent Cloud has been providing persistent block storage services for users with its cutting-edge Cloud Block Storage (CBS). CBS provides highly efficient and reliable persistent block storage services and is widely deployed and used in scenarios such as core database, Content Distribution Network (CDN), and e-commerce systems. To provide customers with high-performance enterprise-level cloud storage services, Tencent Cloud collaborated with Intel to reconstruct and optimize its ultrafast solid-state drive CBS with a brand-new storage engine design with 3rd Gen Intel® Xeon® Scalable processors and Intel® Optane™ persistent memory 200 Series. It has been verified that with better bandwidth, lower latency, and higher Input/output per second (IOPS), the new solution can create an extremely fast cloud storage experience for performance-intensive business scenarios.[3]

Products and Solutions
3rd Generation Intel® Xeon® Scalable Processors
Intel® Optane™ Persistent Memory 200 Series

Industry
Internet

Organization Size
10,001+

Country
China

Learn more
Video
White Paper

1, 2, 3 For more complete information about performance and benchmark results, visit https://www.intel.com/content/www/us/en/customer-spotlight/stories/tencent-cloud-block-storage-customer-story.html

10–20X increase
in the read/write performance of the new single-node engine with persistent memory.[1]

Intel® Optane™ Persistent Memory Powers Baidu's Next-generation User-mode Storage Engine

Baidu's user-mode single-node storage engine is an innovative system for an architecture that separates computing from storage, providing stable and efficient services for Baidu's online and offline products. Based on Intel® Optane™ persistent memory and the Storage Performance Development Kit, the solution meets the data storage challenge of various business lines and achieves storage system reliability, scalability, and high performance with low operating costs. In the near future, Intel® Optane™ persistent memory 200 Series and 3rd Gen Intel® Xeon® Scalable processor will be equipped on Baidu's user-mode single-node storage engine, while system performance will be improved with the new CLWB (Cache Line Write Back) instruction.

"By collaborating with Intel, Baidu greatly improves the performance of a single-node engine through hardware and software collaboration – introducing the Intel® Optane™ persistent memory and Storage Performance Development Kit."

Wang Yanpeng,
Chief Architect,
Infrastructure Division,
Baidu

Products and Solutions
3rd Gen Intel® Xeon® Scalable processors
2nd Gen Intel® Xeon® Scalable processors
Intel® Optane™ persistent memory

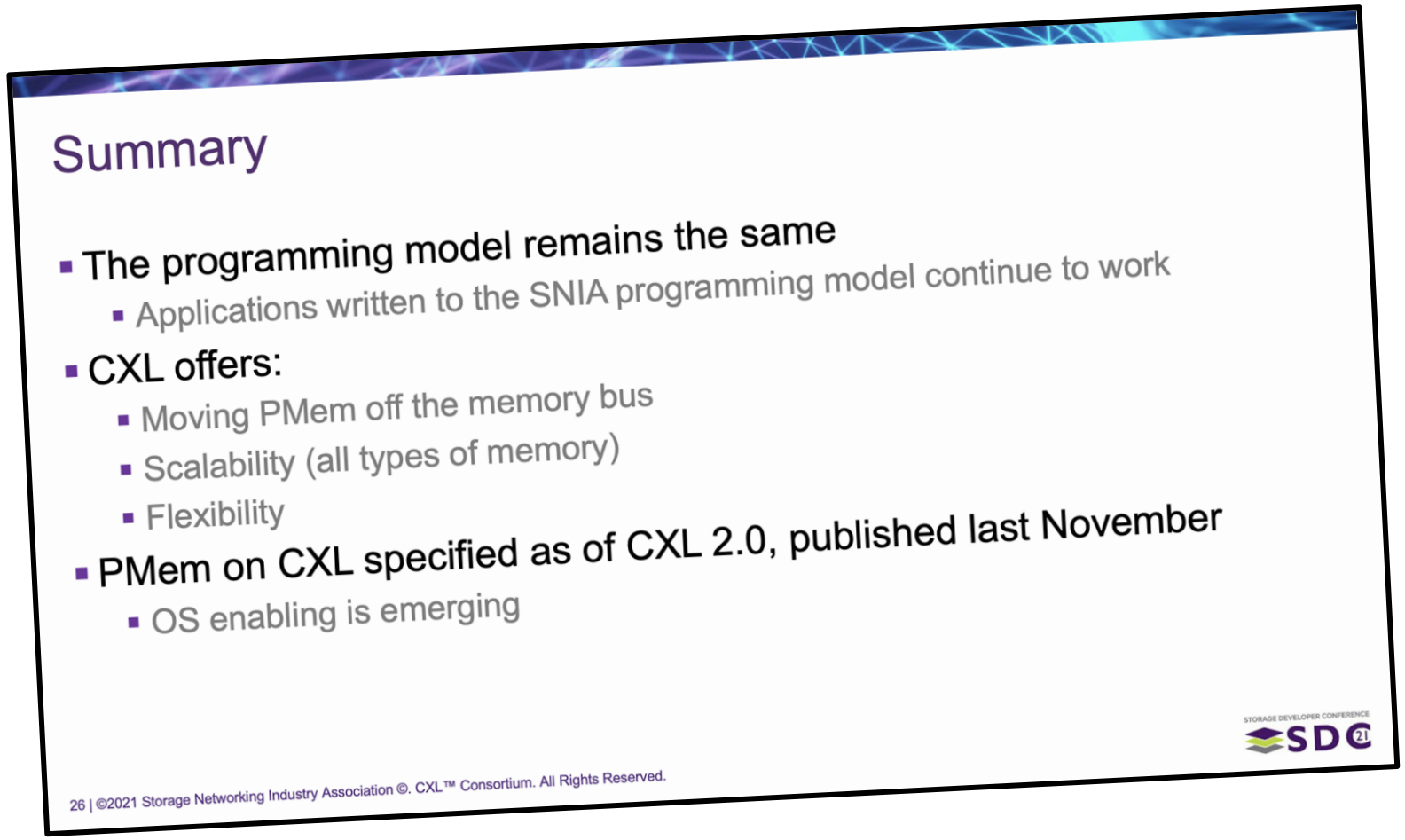
Industry	Organization Size	Country	Learn more
Cloud	10,001+	China	White Paper

1 For more complete information about performance and benchmark results, visit https://www.intel.com/content/www/us/en/customer-spotlight/stories/baidu-optane-spdk-customer-story.html

In this talk…

- The story so far
- A surprising focus for PMem
- How the SNIA PMem Programming Model turned out
- Future directions for PMem

Compute Express Link



- CXL: Very Active!
  - OS enabling underway
  - Active areas like pooling
- CXL + PMem
  - Made for each other

See Alan Benjamin:
Compute Express Link™ CXL™:
Advancing the Next Generation of Data Centers
Tomorrow

Future PMem Directions

- AI/ML
  - Example:
    - Accelerating AI/ML and Data-centric Applications with Temporal Caching
    - https://community.intel.com/t5/Blogs/Tech-Innovation/Artificial-Intelligence-AI/Accelerating-AI-ML-and-Data-centric-Applications-with-Temporal/post/1365954
- More RDMA
  - Examples above
- Metadata persistent store
  - Example: storage appliance metadata
- IMDB
  - The classic use case lives on

See Ziye Yang:
Accelerating Operations on Persistent Memory Device Via Hardware-based Memory Offloading Technique
This Afternoon

Please take a moment to rate this session.

- Your feedback is important to us.

