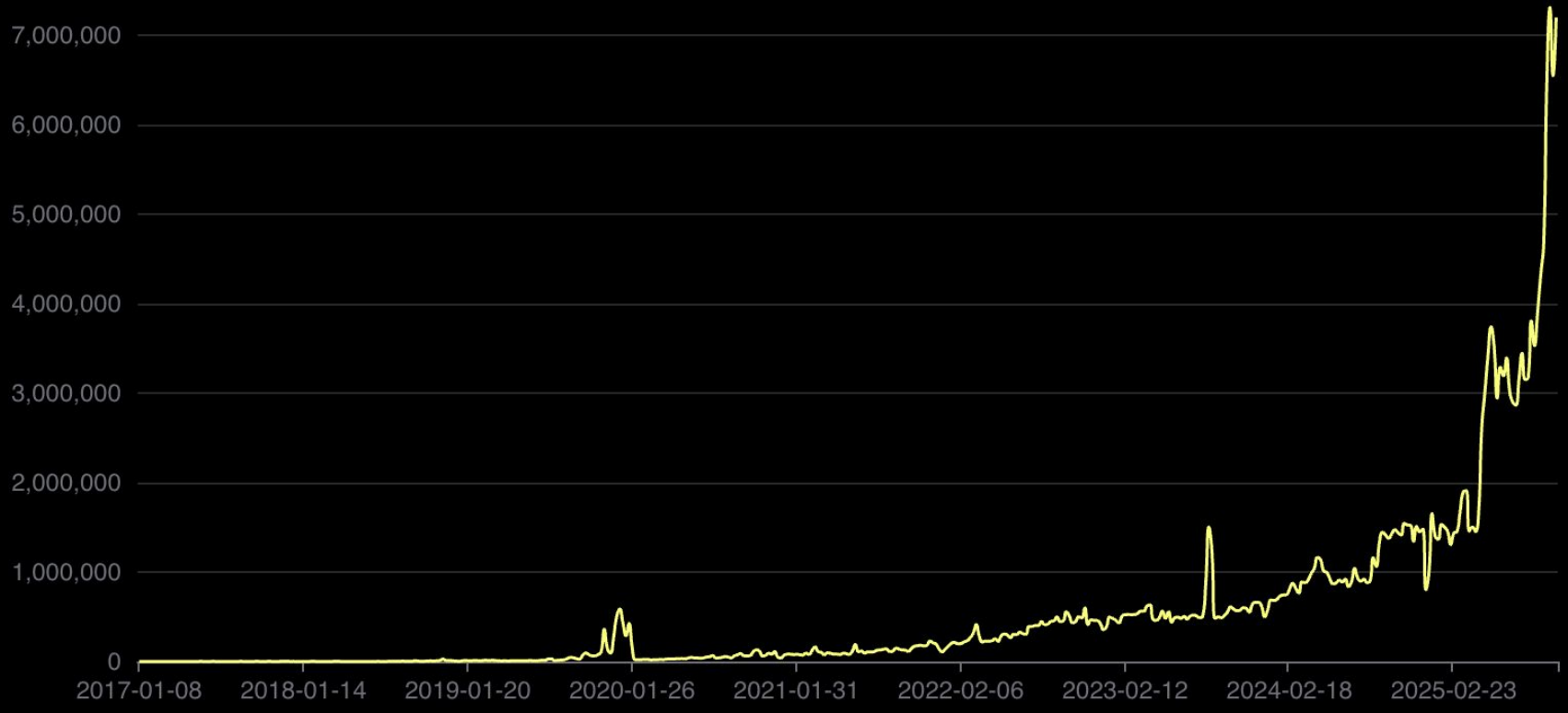
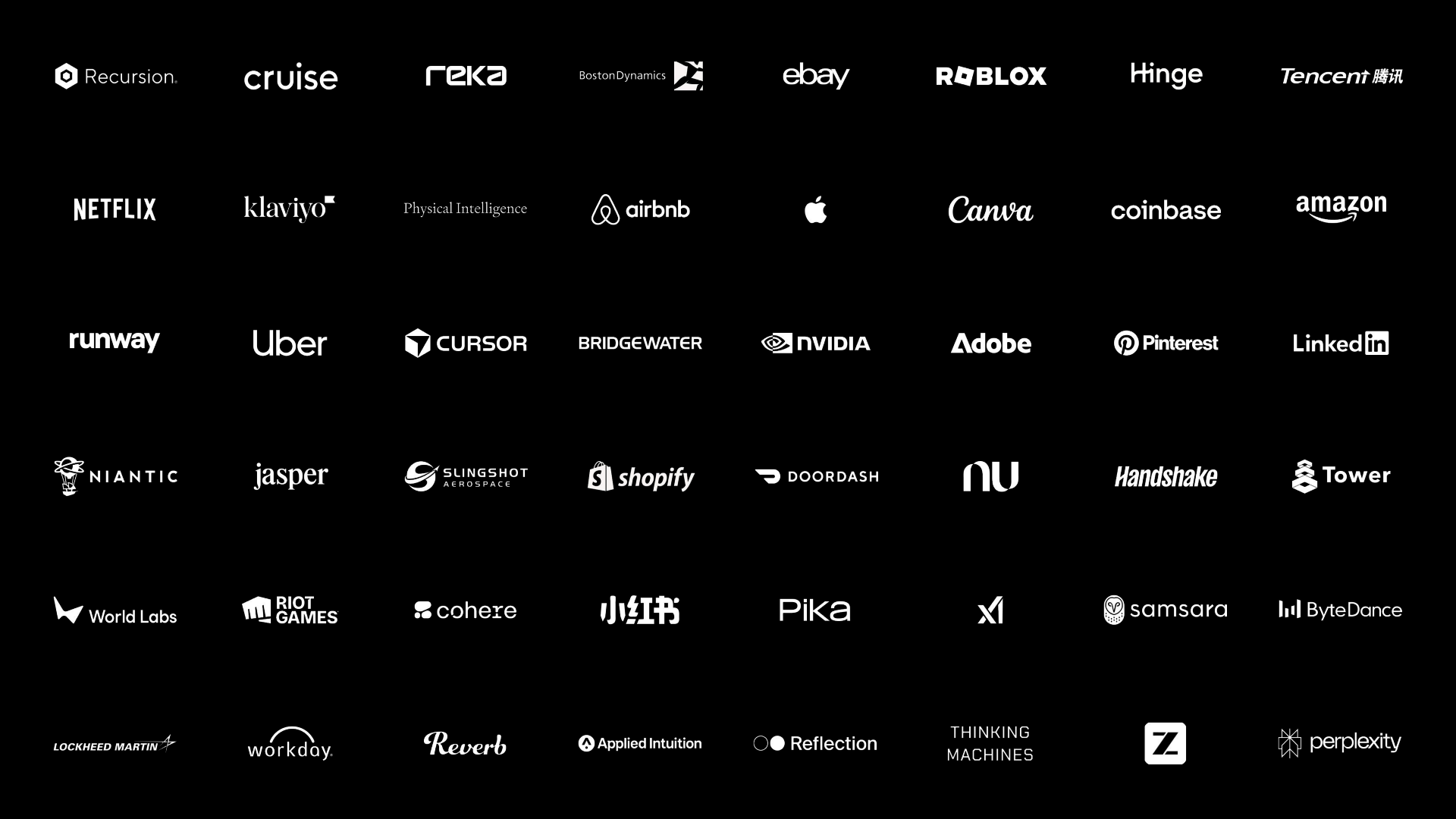
Anyscale

Ray Project Update

Artur Niederfahrenhorst
Member of Technical Staff, Anyscale

Ray was built for the challenges of today




Recursion
cruise
reka
Boston Dynamics
ebay
ROBLOX
Hinge
Tencent 腾讯
NETFLIX
klaviyo
Physical Intelligence
airbnb
Canva
coinbase
amazon
runway
Uber
CURSOR
BRIDGEWATER
NVIDIA
Adobe
Pinterest
LinkedIn
NIANTIC
jasper
SLINGSHOT AEROSPACE
shopify
DOORDASH
Handshake
Tower
World Labs
RIOT GAMES
cohere
小红书
Pika
samsara
ByteDance
LOCKHEED MARTIN
workday.
Reverb
Applied Intuition
Reflection
THINKING MACHINES
perplexity

Ray’s Primitives

Ray's Actor API

```python
class Trainer:
    def train(self):
        # Training step




class Generator:
    def generate(self):
        # Generate
```

Ray's Actor API

```python
class Trainer:
    def train(self):
        # Training step

    def get_weights(self):
        return self.model.state_dict()


class Generator:
    def generate(self):
        # Generate

    def set_weights(self, weights):
        # Set weights
```

Ray's Actor API

```python
@ray.remote(accelerator_type="H100")
class Trainer:
    def train(self):
        # Training step

    def get_weights(self):
        return self.model.state_dict()


@ray.remote(accelerator_type="B200")
class Generator:
    def generate(self):
        # Generate

    def set_weights(self, weights):
        # Set weights
```

Ray's Actor API

```python
@ray.remote(accelerator_type="H100")
class Trainer:
    def train(self):
        # Training step

    def get_weights(self):
        return self.model.state_dict()


@ray.remote(accelerator_type="B200")
class Generator:
    def generate(self):
        # Generate

    def set_weights(self, weights):
        # Set weights
```

```python
# Set up actors
trainer = Trainer.remote()
generator = Generator.remote()
```

Ray's Actor API

```python
@ray.remote(accelerator_type="H100")
class Trainer:
    def train(self):
        # Training step

    def get_weights(self):
        return self.model.state_dict()


@ray.remote(accelerator_type="B200")
class Generator:
    def generate(self):
        # Generate

    def set_weights(self, weights):
        # Set weights
```

```python
# Set up actors
trainer = Trainer.remote()
generator = Generator.remote()

while True:
    # Train
    trainer.train.remote()

    # Synchronize weights
    ref = trainer.get_weights.remote()
    generator.set_weights.remote(ref)

    # Rollouts
    generator.generate.remote()
```

Ray Direct Transport (alpha) - Native RDMA support

```python
@ray.remote(accelerator_type="H100")
class Trainer:
    def train(self):
        # Training step

    @ray.method(tensor_transport="nixl")
    def get_weights(self):
        return self.model.state_dict()


@ray.remote(accelerator_type="B200")
class Generator:
    def generate(self):
        # Generate

    def set_weights(self, weights):
        # Set weights
```

```python
# Set up actors
trainer = Trainer.remote()
generator = Generator.remote()

while True:
    # Train
    trainer.train.remote()

    # Synchronize weights
    ref = trainer.get_weights.remote()
    generator.set_weights.remote(ref)

    # Rollouts
    generator.generate.remote()
```

GPU object transport over RDMA

Nearly every open-source RL framework is built on Ray

Framework	Training engine	Serving engine	Orchestrator
TRL (Hugging Face)	Hugging Face Trainer	Hugging Face, vLLM	—
Verl (ByteDance)	FSDP, DeepSpeed, Megatron	vLLM, SGLang	Ray
OpenRLHF	DeepSpeed	Hugging Face, vLLM	Ray
AReaL (Ant Group)	DeepSpeed, Megatron	vLLM, SGLang	Ray
Prime RL	FSDP	vLLM	—
ROLL (Alibaba)	DeepSpeed, Megatron	vLLM, SGLang	Ray
NeMo-RL (Nvidia)	FSDP, Megatron	vLLM	Ray
SkyRL (UC Berkeley)	FSDP, DeepSpeed	vLLM, SGLang, OpenAI	Ray
SLIME (Z.ai)	Megatron	SGLang	Ray
RAGEN	Verl backend	Hugging Face, vLLM, SGLang	Ray

AI Data Pipelines

Nvidia uses Ray for AI Data pipelines

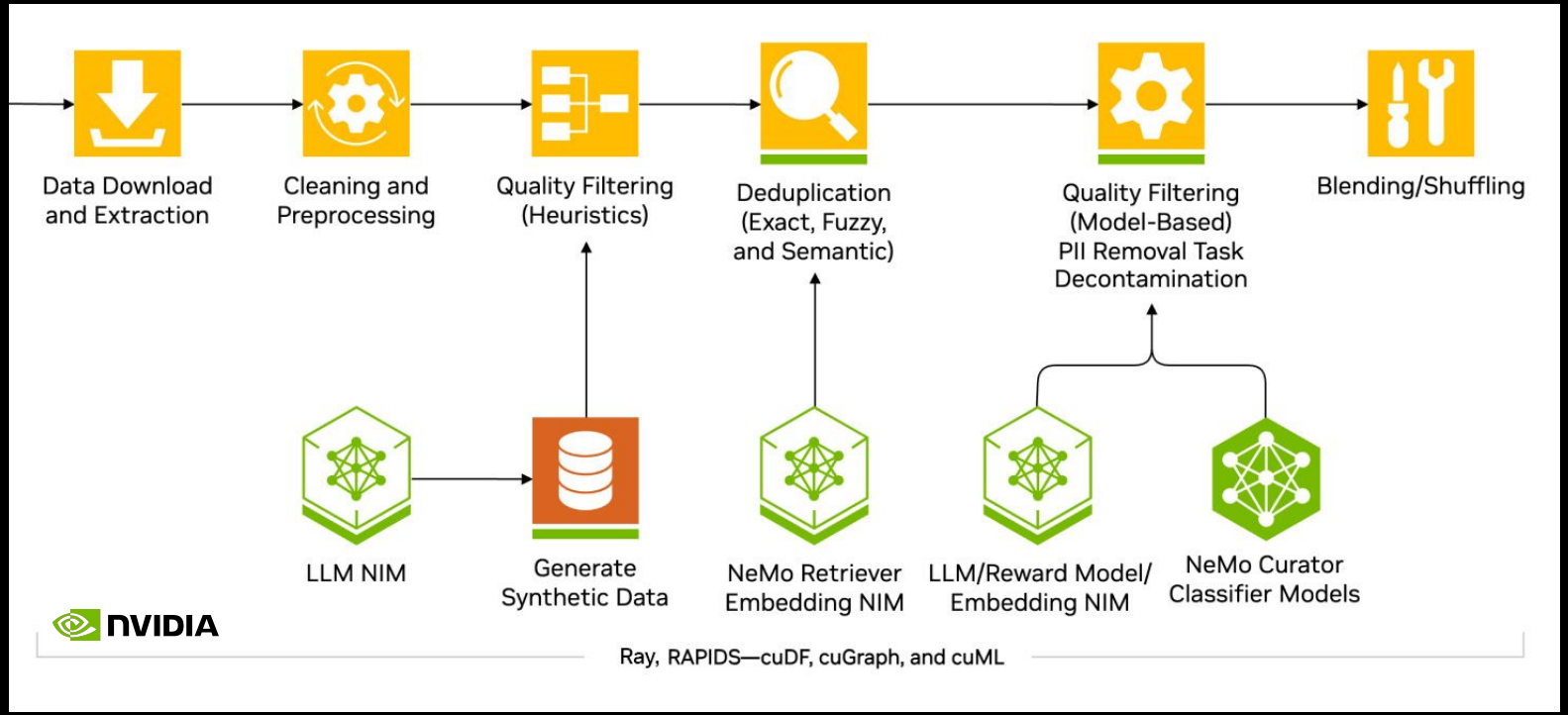


Image from developer.nvidia.com/nemo-curator

The ecosystem around RAY

AIBrix ByteDance	SLIME Z	OpenRLHF OpenRLHF	AReaL ANT GROUP	Syftr DataRobot
SkyRL Berkeley	Data-Juicer Alibaba.com	NeMo Curator NVIDIA	ROLL Alibaba.com	NeMo-RL NVIDIA
RayDP intel	Cosmos Curate NVIDIA	DeltaCAT amazon	Daft eventual	verl ByteDance

Thank you!