第2部 AI最前線

1. AIインフラ：サービス提供者側からの技術展望

Internet Week 2025

2025/11/27

小林 正幸

SAKURA internet

自己紹介

小林 正幸（Masayuki Kobayashi）

さくらインターネット インフラエンジニア

担当業務：

- ベアメタル型GPUクラウドサービス「高火力 PHY」のインフラ設計・構築
- 先端技術の調査・検証

略歴：

- 2025-現在　さくらインターネット
- 2017-2024　LINE → ヤフー → LINEヤフー
- 2014-2017　IIJ

会社概要

データセンターを中心とした各種クラウドサービスを提供
社会と顧客のDXを支援

1996 さくらインターネット創業
1996年12月に現社長の田中邦裕が、
舞鶴高専在学中に学内ベンチャーとして創業

1999 株式会社を設立 / 最初のデータセンター開設
1999年8月に株式会社を設立。10月には、第1号となる
データセンターを大阪市中央区に開設

2005 東証マザーズ上場
2005年10月に東京証券取引所マザーズ市場に上場

2011 石狩データセンター開設
2011年11月、北海道石狩市に国内最大級の
郊外型大規模データセンターを開設

2015 東証一部に市場変更
2015年11月に東京証券取引所市場第一部に市場変更

2021 創業25周年
2021年12月、創業25周年

2022 東証プライム市場へ移行
東京証券取引所 新市場区分のプライム市場へ移行

本社所在地	大阪府大阪市北区大深町6-38 グランフロント大阪 北館 JAM BASE 3F
創業年月日	1996年12月23日
設立年月日	1999年8月17日
資本金	112億8,316万円
上場年月日	2005年10月12日（マザーズ） 2015年11月27日（東証一部） 2022年4月（東証プライム）
従業員数	連結 997名（2025年3月末）

石狩データセンター

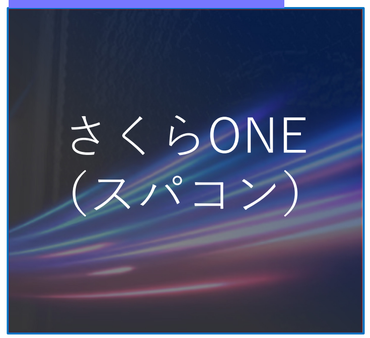
国内基盤と技術力で支える、GPUクラウドサービス

国産デジタルインフラへの期待が高まる中、さくらインターネットはこれまで培ってきた国内運用基盤と技術力を活かし、
学習から推論まで幅広いニーズに応える高付加価値GPUクラウドサービスを展開。

学習（モデル開発）向け　多様なニーズに答えるサービスラインアップ　推論（モデル利用）向け

一覧

マネージドスパコン



高火力シリーズ

物理サーバを提供

仮想マシンを提供

コンテナ実行基盤を提供

さくらのAI

AI Engine

AIソリューション

主な用途

学習

自社スパコン（クラスタ）とベアメタルで
新型GPU「NVIDIA B200」を早期提供。
大規模分散学習のニーズにも対応。

推論

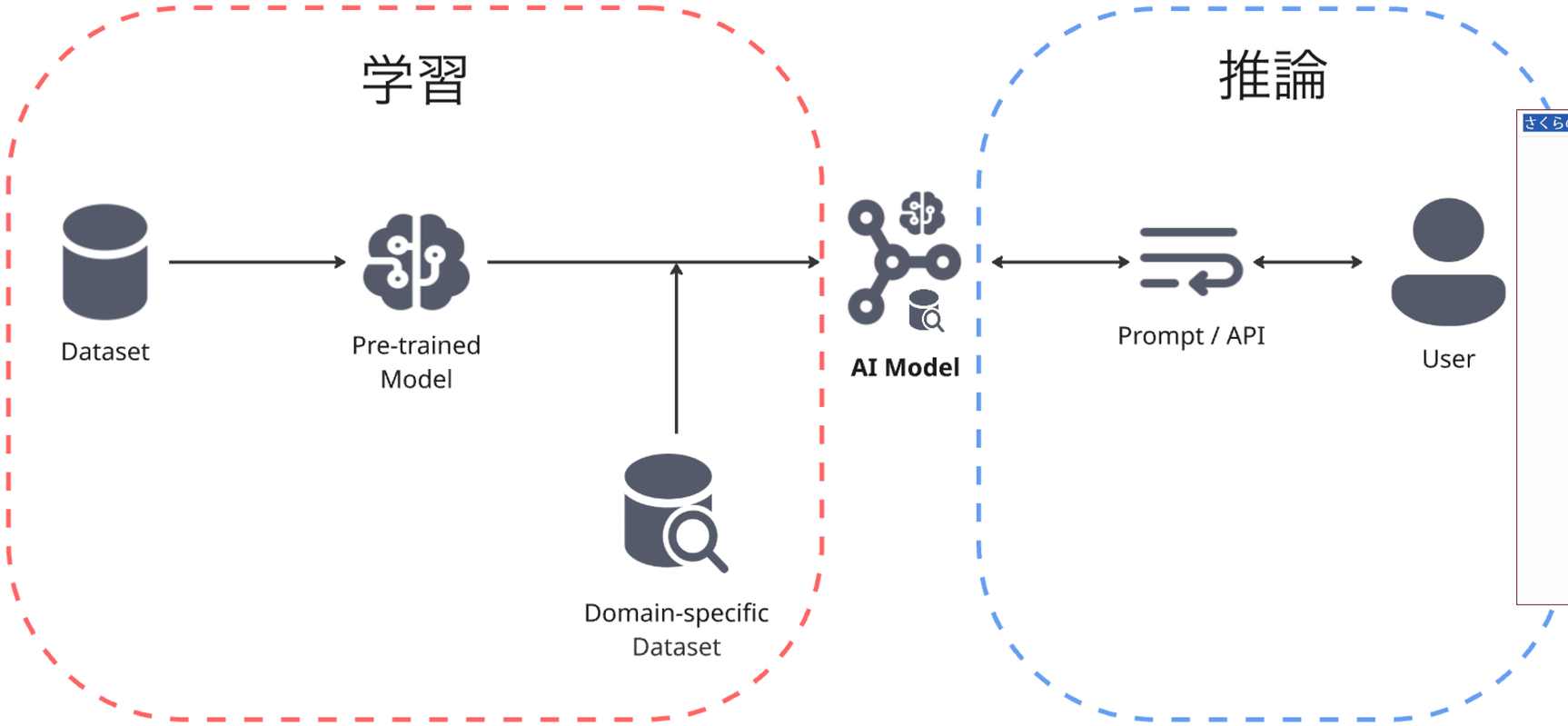
GPUが接続されたVMやコンテナをクラウド上で
手軽に実行可能。
AIアプリケーションの開発・運用を効率化。

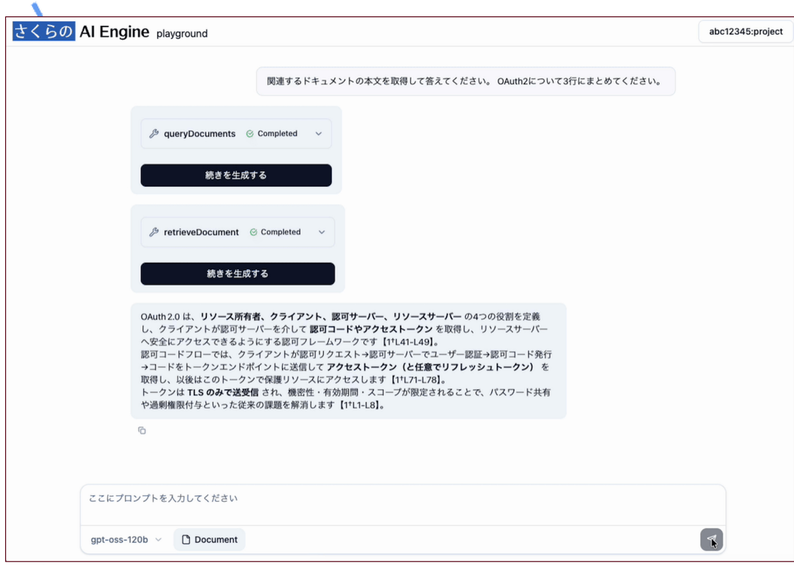
最初に

- このセッションではAIワークロードを支えるインフラ技術の最先端について、ネットワーク技術者の観点で可能な限りその本質と展望にフォーカスします。
- 技術動向含めすべてを紐解くことは時間的に不可能なため、要素技術の詳細は関連資料を適宜ご参照ください
  - AIインフラを考える
    - https://speakerdeck.com/markunet/aiinhurawokao-eru
  - GPUクラスタネットワークとその設計思想
    - https://techblog.lycorp.co.jp/ja/20250115a
  - GPUネットワーク設計・運用 基礎勉強会 Lossless Ethernet - PFC/ECN編
    - https://speakerdeck.com/markunet/ecnbian
  - AI時代のデータセンターネットワーク -第40回 情報ネットワーク・ネットワークシステム研究ワークショップ
    - https://speakerdeck.com/lycorptech_jp/dcnw_in_the_ai_era
  - EthernetベースのGPUクラスタ導入による学びと展望 - NVIDIA AI Summit Japan 2024
    - https://speakerdeck.com/lycorptech_jp/20241202
  - Podcast: fukabori.fm "124. AI時代のGPUクラスタ、DCネットワーク"
    - https://fukabori.fm/episode/124

AIインフラ == 生成AIのためのインフラ

- モデルの学習と推論サービスのための基盤
- **学習: 大規模なデータセットで分散学習を行い、（事前）学習済みモデルを作成する**
- **推論: 学習済みモデルをデプロイしてユーザーリクエストを処理する**





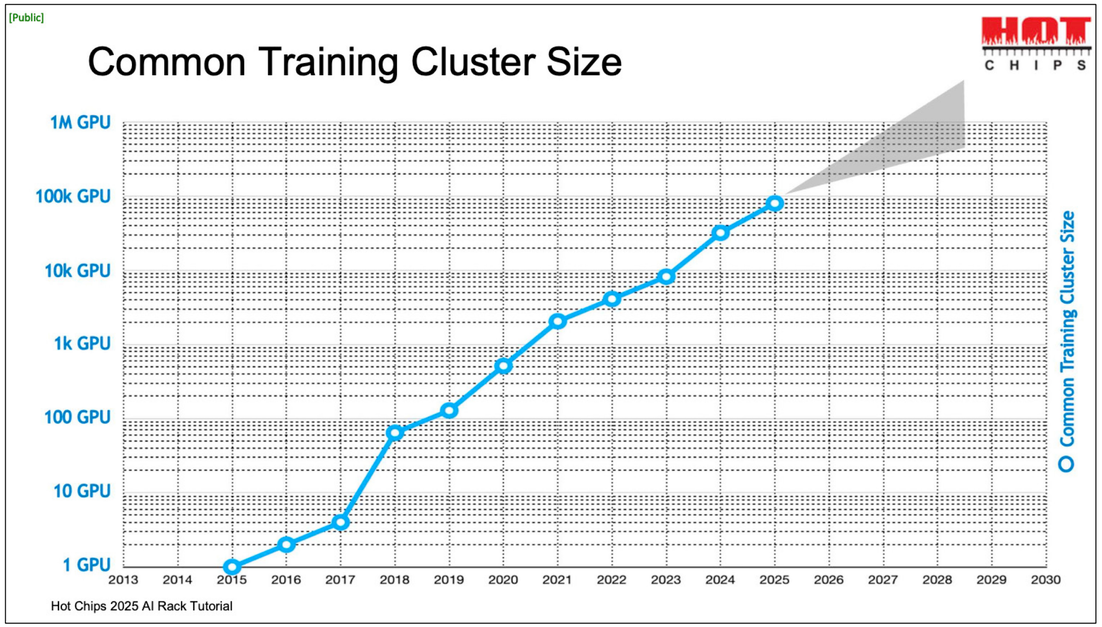
分散学習のためのインフラ

学習のための基盤

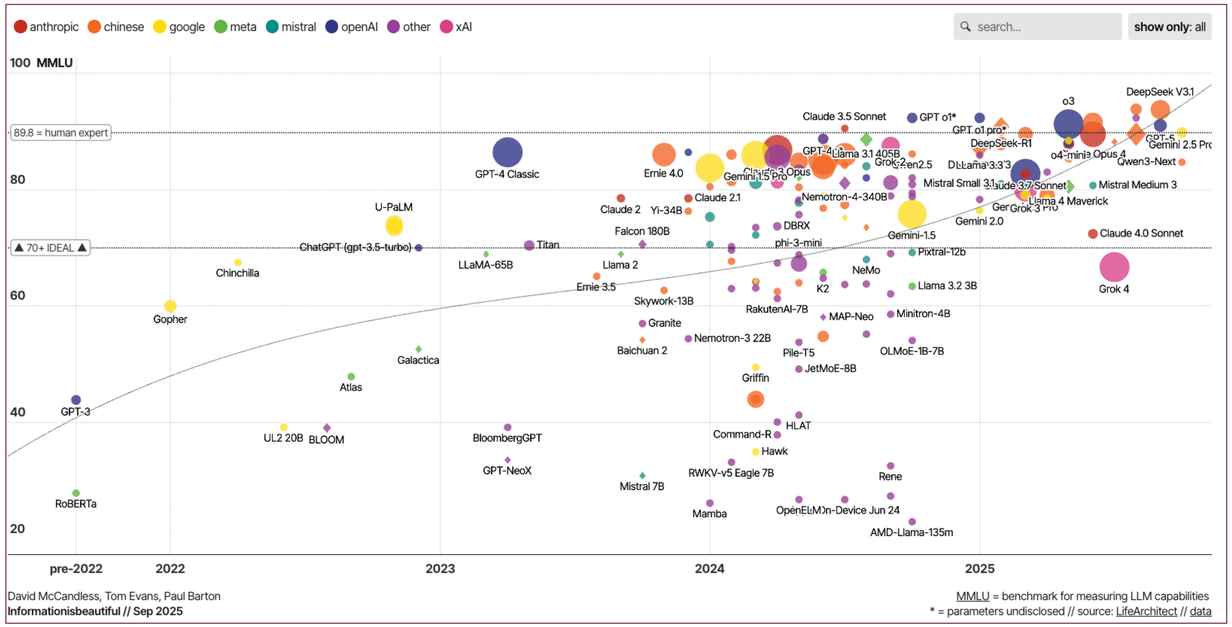
- 高度な推論性能を持つフロンティアモデルは巨大化の傾向にある
- 大規模モデルはデータ（パラメータ）の量を増やすと推論性能が向上する。この法則は現在も有効。

  → **学習（モデル開発）のための計算・通信コスト**とエネルギーコストが急激に増加

※ 大規模モデルの性能はパラメータ数の規模に応じて変化するが、その関係は線形ではない



“The Hot Chip is a Rack” (AI Literally Demands we Think Outside the Box)
https://hc2025.hotchips.org/



Major Large Language Models (LLMs) ranked by capabilities, sized by billion parameters used for training
https://informationisbeautiful.net/visualizations/the-rise-of-generative-ai-large-language-models-llms-like-chatgpt/

学習とネットワーク

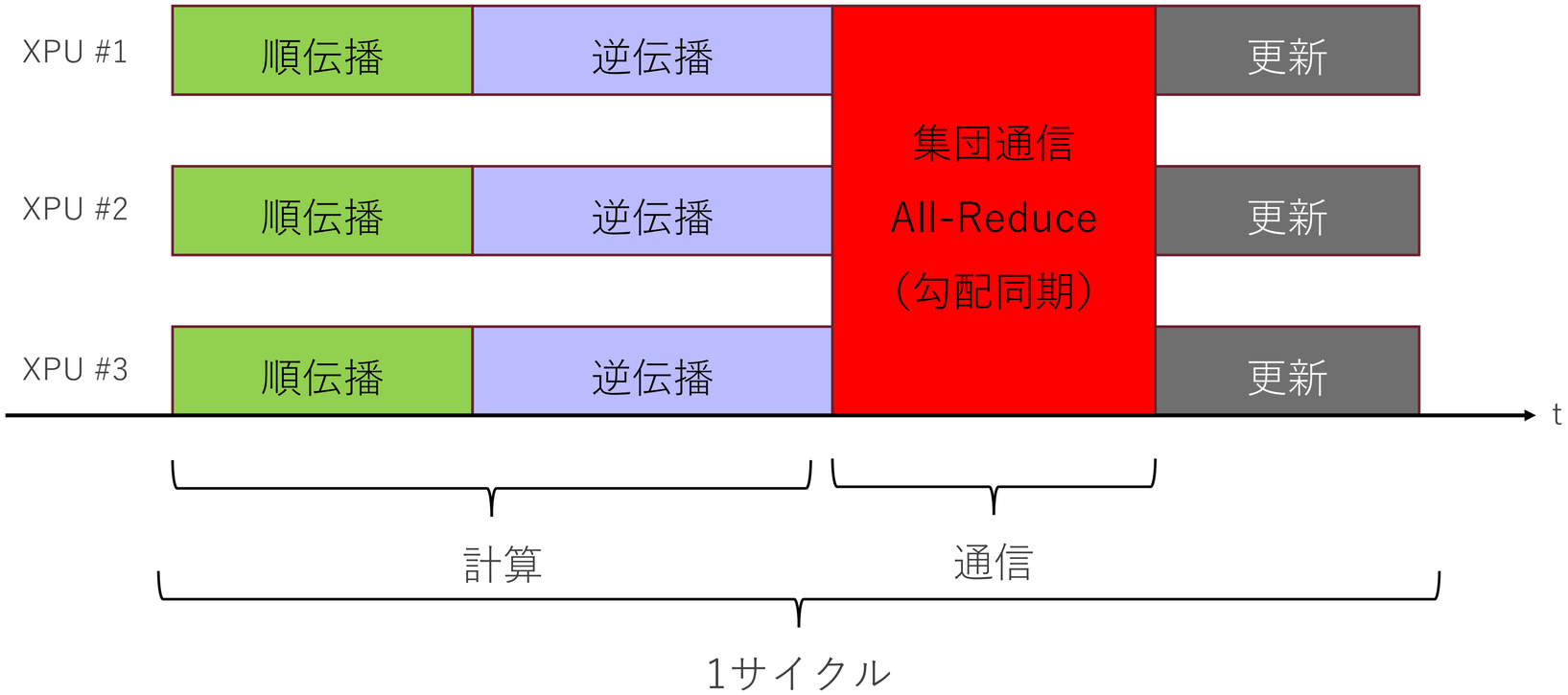
- 最近の大規模モデルはそもそも「1台のXPUのメモリに載らない」
- 大規模モデルのメモリ消費は主に以下の合算 → 計算でもメモリを消費する
  - パラメータ（重み）：例）BF16なら 2 byte/param
  - オプティマイザ状態
  - アクティベーション（順伝播の中間値）：シーケンス長やバッチサイズに比例
  - KV Cache（推論時）：シーケンス長×層×ヘッド次元に比例
- 例：70Bモデル（BF16 2 byte）
  - 重み：70×10^9×2 ≈ 140 GB
  - オプティマイザ状態：実装次第で数百GB規模
  - これだけで現行世代XPUのHBMを圧迫し、単一XPUに載せることは難しい

必然的に **複数XPUでの分割（並列化）** が必要

分割するということはデータの同期・更新のために**通信が必要** → 集団通信

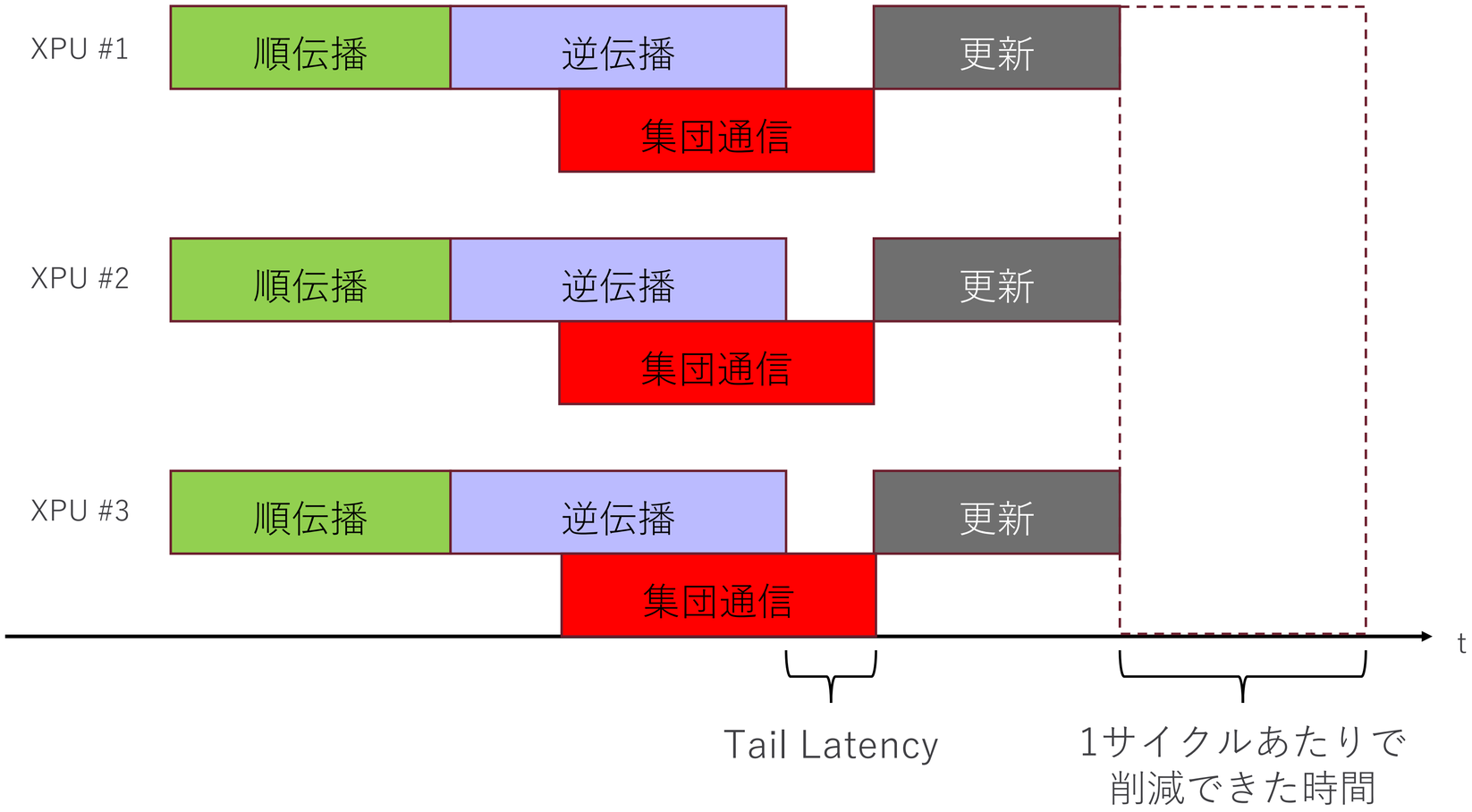
学習とネットワーク

- 学習は 順伝播 → **逆伝播 （勾配同期）** → パラメータ更新 の順で進む
  - 計算と通信が繰り返し実行される
- 順伝播ではデータをニューラルネットワークに通して予測計算し、誤差（損失）を計算
- 逆伝播では損失のパラメータによる微分を計算
  - ここで得られた結果が「勾配」と呼ばれるもの → どれだけ重みを動かすべきかという情報
  - 勾配を使って損失が小さくなるようにパラメータ（重み）を更新する



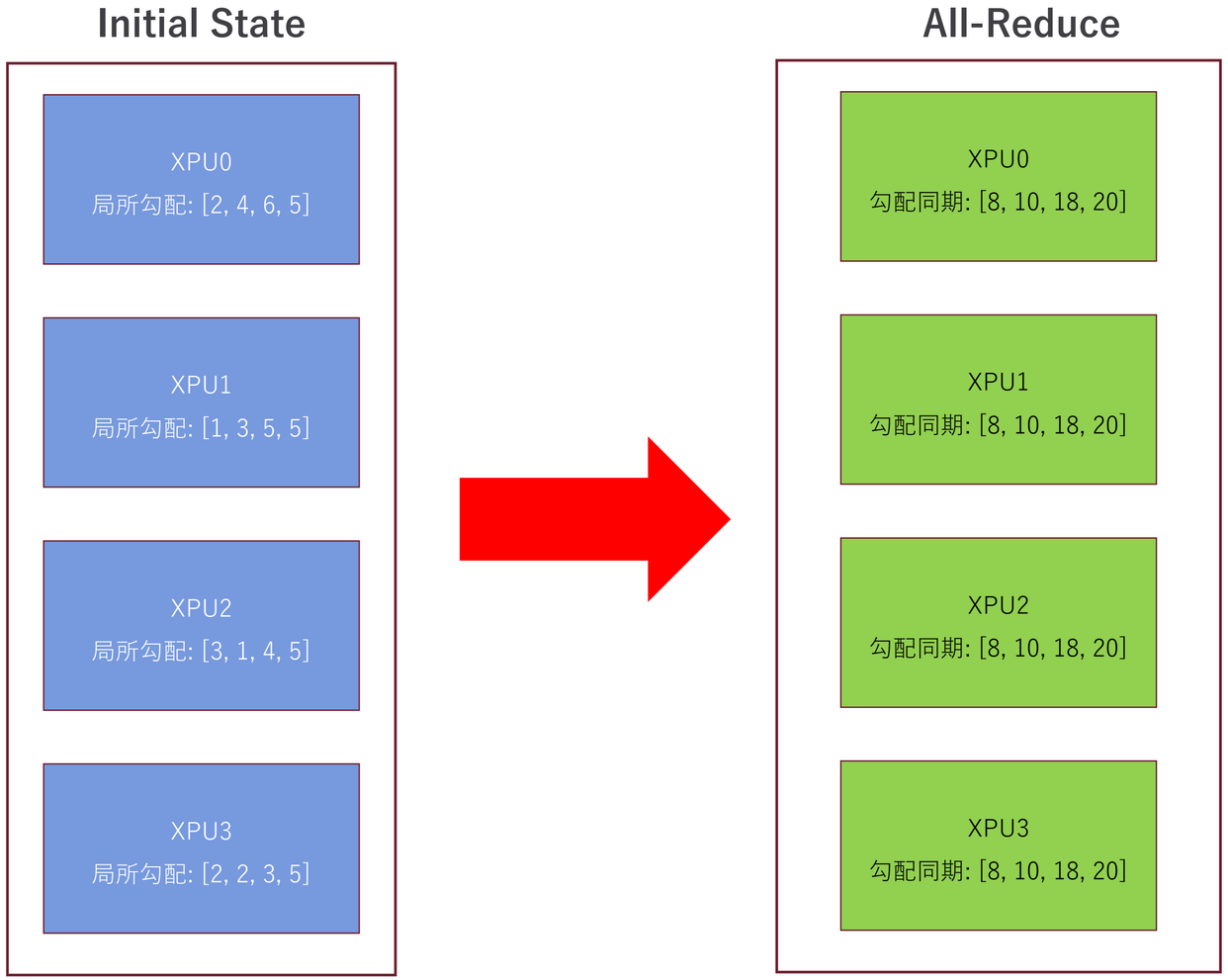
なぜ通信が速い必要があるのか？

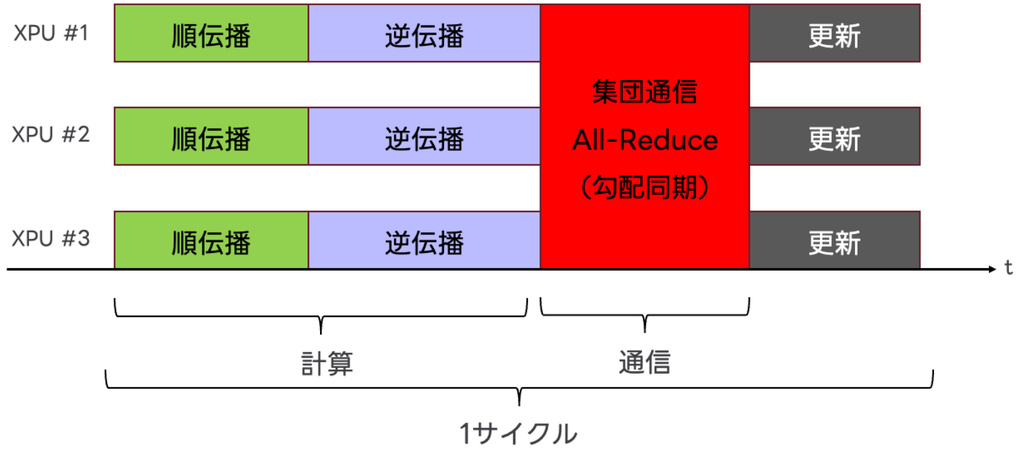
- 学習の速度（効率）を上げるためには、**計算の裏に通信を隠蔽（オーバーラップ）することが重要**
- 逆伝播計算からはみ出た通信時間がTail Latencyとして完了時間に影響する → 速いことが求められる
- **オーバーラップは学習のパフォーマンス向上の本質** → Scale Across (DC間の長距離分散学習)でも有用



なぜ通信が速い必要があるのか？

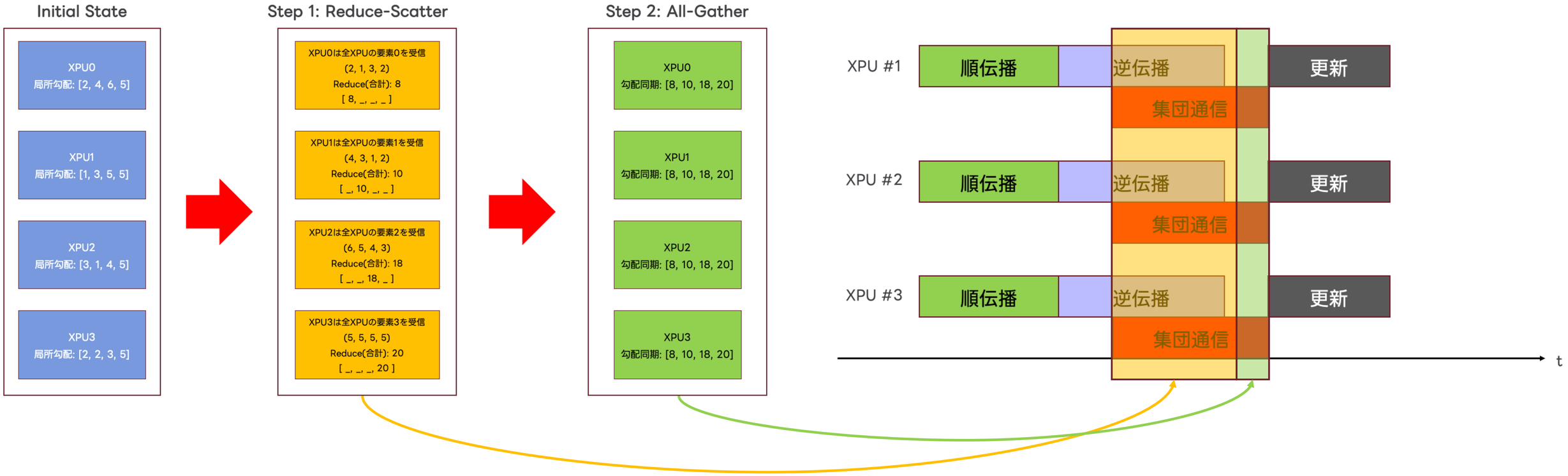
- オーバーラップしない場合、計算がすべて完了してから集団通信（All-Reduce）になるため遅い
- 通信している間はXPUは計算していないため、計算資源として無駄になってしまう





なぜ通信が速い必要があるのか？

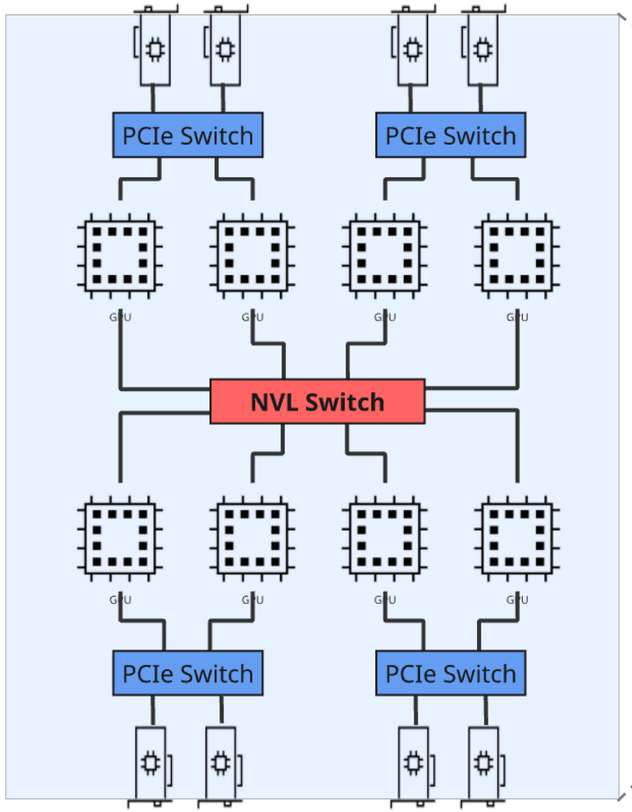
- Reduce-Scatter + All-Gather = All-Reduce の例
- 逆伝播中、勾配がバケット単位で準備できた瞬間から通信を非同期に開始
- 計算が完了した部分から勾配の同期を行える → Reduce-Scatter部分をオーバーラップ
- オーバーラップできる通信が速いほど、逆伝播計算からはみ出る時間が短くなる



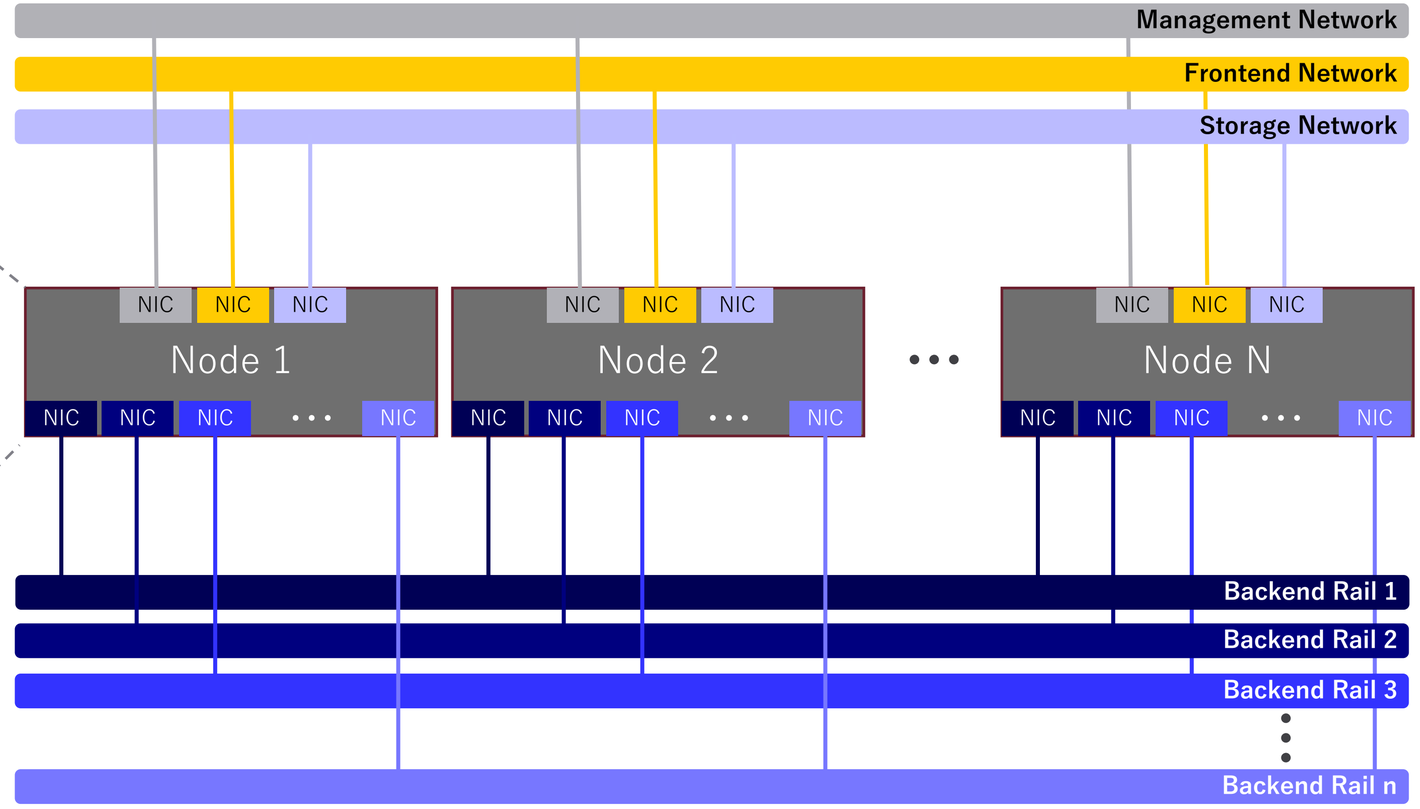
データパスとトポロジ

- XPUとメモリ、ストレージ間でやり取りされるデータの通り道を意識する
- ノード内のNVLinkやPCIeのデータパスを「**Scale Up**」と呼び、ノード間を「**Scale Out**」と呼ぶ
- Scale Outのトポロジは Scale Upを効率的（低遅延）にノード外に延伸しようとした結果生まれたもの

→ **Scale Up が本質的に重要な部分**



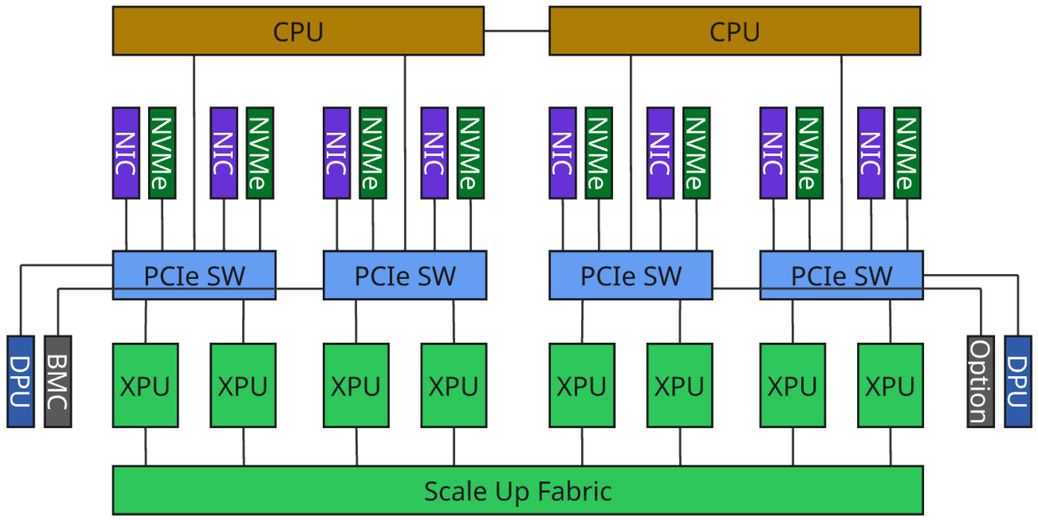
Scale Up



Scale Out

データパスとトポロジ

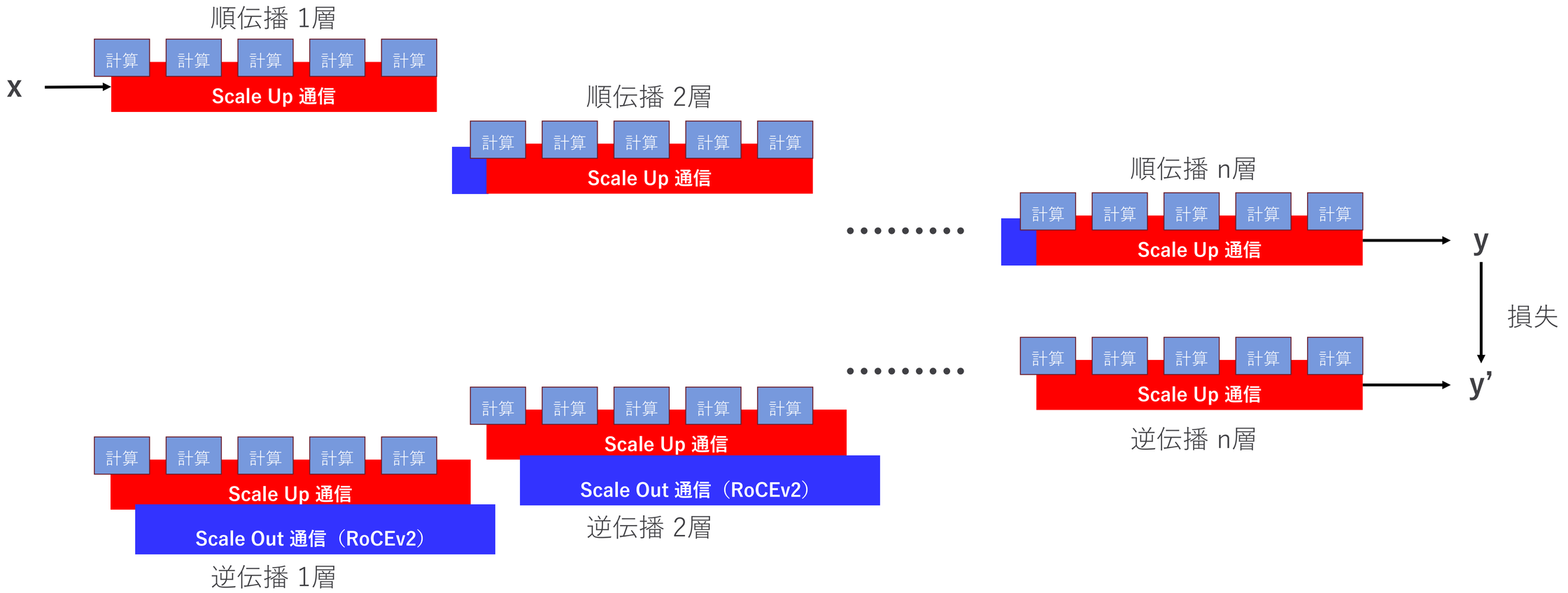
- Scale UpはScale Outより高帯域・低遅延
  - サーバの内部レイアウトやインフラ構成によって選択できる経路が変わる → **どこがボトルネックになるか知る**
  - データパスがCPUを経由しないか、NUMAを跨がないか、PCIeのレーン数は何本あるかなど
- 遅延に敏感なアプリケーションはなるべく最短経路の Scale Up でデータパスが完結するようにする
  - 特にストレージや大規模な推論基盤などの設計で重要になる



ドメイン	インターコネクト	最大伝送速度（1XPUあたり）	備考
Scale Up	PCIe Gen5	64GB/s (x16,単一方向)	1レーンあたり4GB/s
	PCIe Gen6	128GB/s (x16,単一方向)	1レーンあたり8GB/s
	NVLink 4.0	450GB/s (18Links,単一方向)	Hopper 1リンクあたり25GB/s
	NVLink 5.0	900GB/s (18Links,単一方向)	Blackwell 1リンクあたり50GB/s
Scale Out	400G NIC/DPU	50GB/s (双方向)	ConnectX-7 / BlueField-3
	800G NIC	100GB/s (双方向)	ConnectX-8

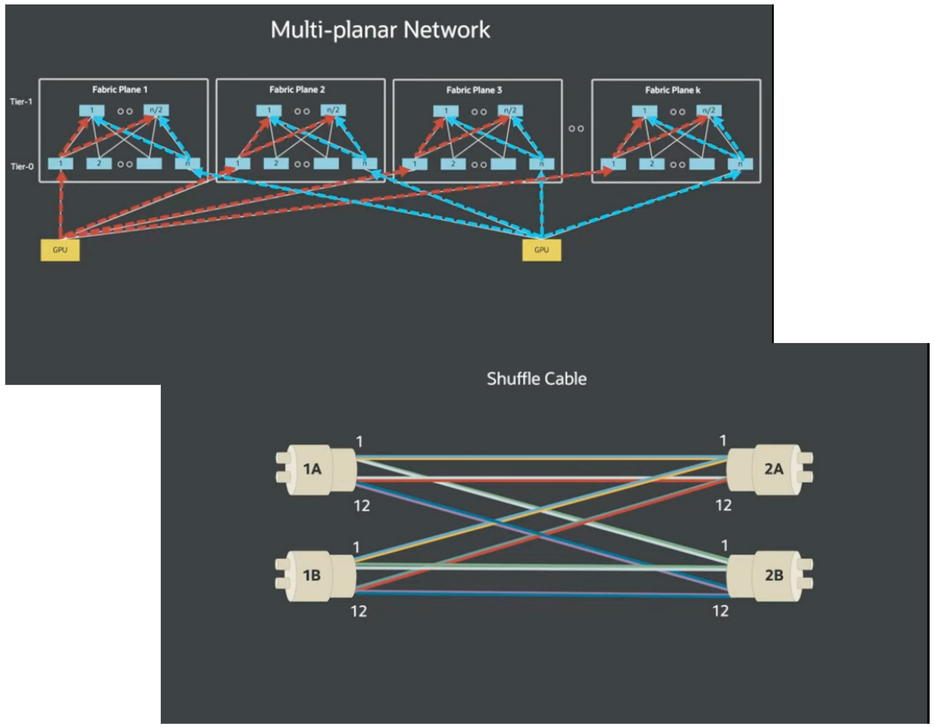
学習においてどのネットワークがいつ使用されるか

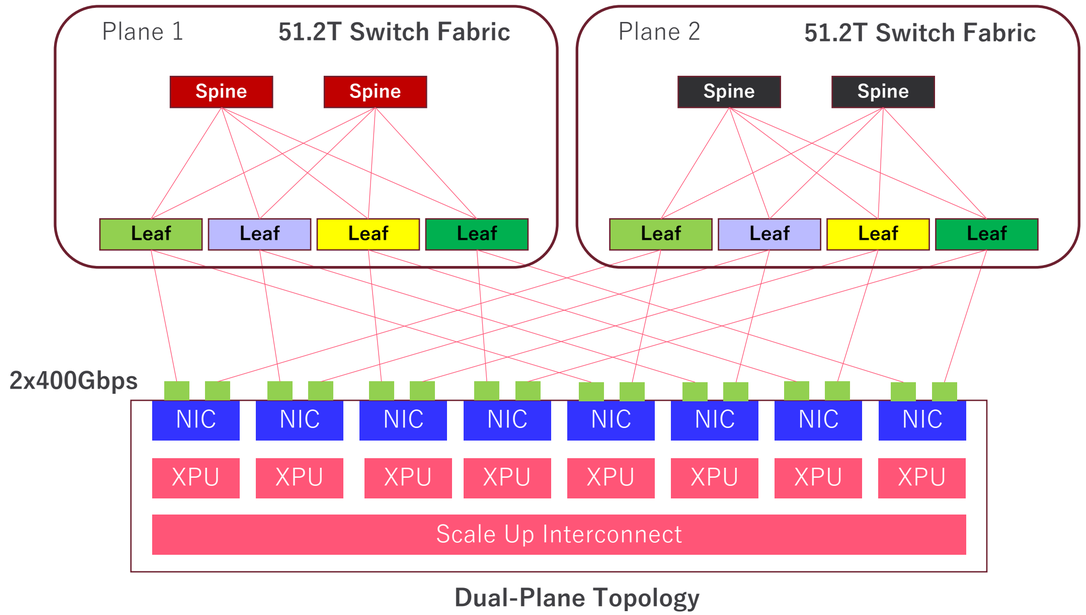
- 分散学習（Pre-training）では逆伝播の際に膨大な通信が発生する
- 計算の裏に通信をオーバーラップ（隠蔽）するために高速なScale Outが必要



学習基盤におけるScale Outの今後の展望

- NICの高速化を推し進めている状況は継続、800Gは市場に出回り、1.6Tも見えている
- 高速化によって僅かなパケットロスやエラーの影響がよりシビアになりつつある
  - 高速な再送要求をトリガーできるトランスポートや、リンクレベルの再送制御技術の発展に期待（検証可能な実装待ち）
- レジリエンスとRadix（収容可能なXPU数）を向上させるマルチプレーンの物理構成などに注目



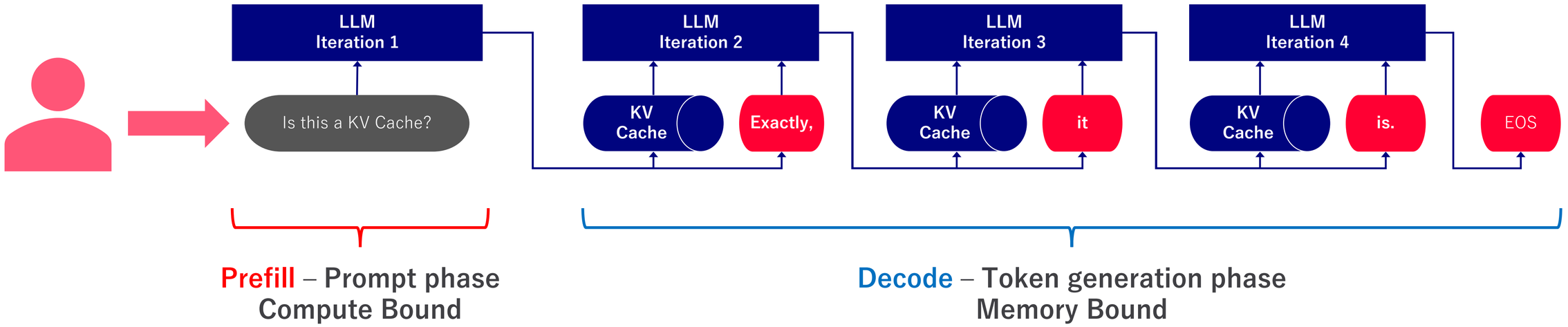


First Principles: Oracle Acceleron Multiplanar Network Architecture
https://blogs.oracle.com/cloud-infrastructure/post/first-principles-oracle-acceleron-multiplanar

（分散）推論のためのインフラ

推論のための基盤 - LLMの推論とは

- 推論はユーザーの入力を元に、モデルが次の単語を予測し（そのトークンを）生成する
- さらにその生成したトークンを系列の末尾に加え、再度次のトークンの予測計算を自己回帰的に行う
  - 新しいトークン以外はただの再計算になる → キャッシュして使い回せば効率的（**KV Cache**）
- 推論は Prefill と Decode と呼ばれる異なる2つのステップから構成される
  - **Prefill**：最初の出力トークンを生成するフェーズ、同時にKV Cacheも生成、行列演算で計算リソースを消費
  - **Decode**：直前のトークンを系列に加え、後続の出力を生成していくフェーズ。
    KV Cacheを更新していくためメモリ操作がボトルネックになる。



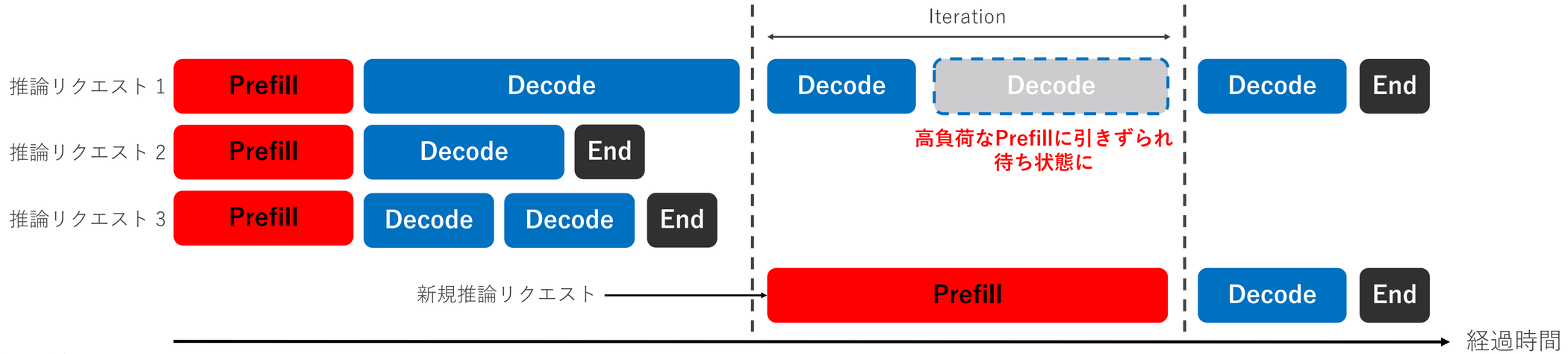
推論基盤の何が難しいのか？

- 一般的なWebサービスのリクエスト・レスポンスと異なり、**推論は「超・不定形ワークロード」**

  - 入力内容とその長さ
  - 出力内容とその長さ

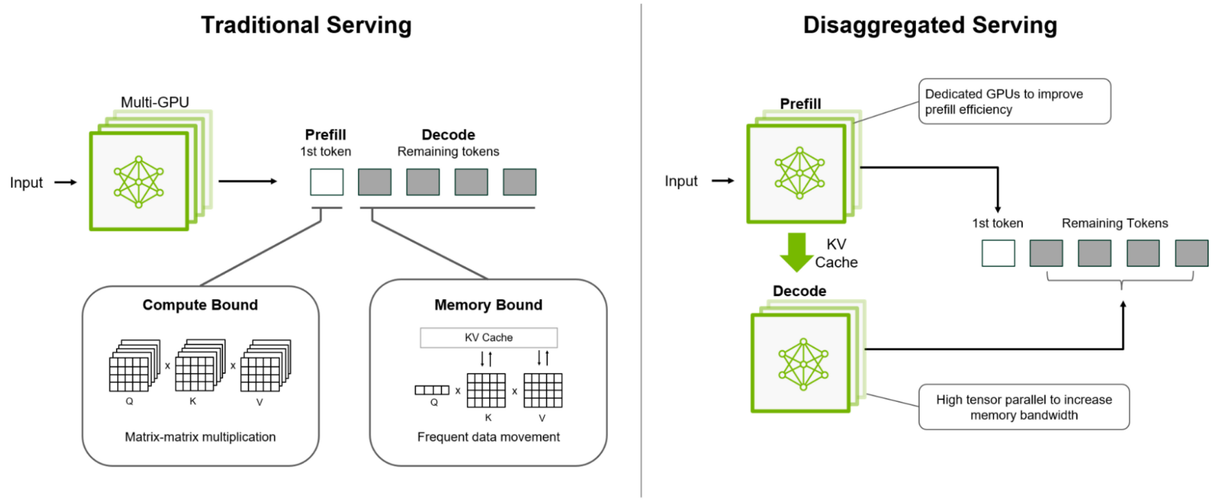
  がユーザーごとに大きく異なり、到着タイミングもランダムで同時に処理するリクエスト数にも影響される。

- 推論基盤の負荷はユーザーの入力傾向に強く依存し、不確実性が高く最適化が難しい
  - PrefillとDecodeが同一バッチに混ざることで、一問一答のような短出力のユーザーリクエストが、出力の長い高負荷なユーザーリクエストに引きずられてしまい、レイテンシを悪化させることがある。



PD Disaggregation

- 特性とボトルネックの異なる**PrefillとDecodeを別々のWorker（XPU）に分離**する
  - **PD Disaggregation** → 行列演算（Prefill）がメインのXPUと、メモリアクセス（Decode）がメインのXPUを分ける
  - PrefillとDecodeが同一バッチで処理されることによる問題を回避可能
- 分離したことで新たな課題も生まれる
  - **KV CacheをWorker間で転送する必要がある → 転送に必要なキャッシュサイズとネットワークが鍵**
  - キャッシュの使用率などを見ながら、リクエストを適切なキャッシュにルーティングすること



NVIDIA Dynamo, A Low-Latency Distributed Inference Framework for Scaling Reasoning AI Models
https://developer.nvidia.com/blog/introducing-nvidia-dynamo-a-low-latency-distributed-inference-framework-for-scaling-reasoning-ai-models/

KV Cache サイズとデータ転送の見積もり

		Llama3 8B	Llama3 405B
KV Cache サイズ	n_layers	32	126
	n_heads	8	8
	head_dim	128	128
	KV Cache / layer = 2 × (n_heads × head_dim) × 2 [FP16]	4,096	4,096
	入力トークン数 = 1K		
	KV Cache size [GB]	0.13	0.5
	入力トークン数 = 8K		
	KV Cache size [GB]	1.0	4.0
	入力トークン数 8K × 同時接続リクエスト数 100		
	KV Cache size [GB]	100	400
KV Cache 転送時間	100Gbps	8.0s	32.0s
	400Gbps	2.0s	8.0s
	800Gbps	1.0s	4.0s

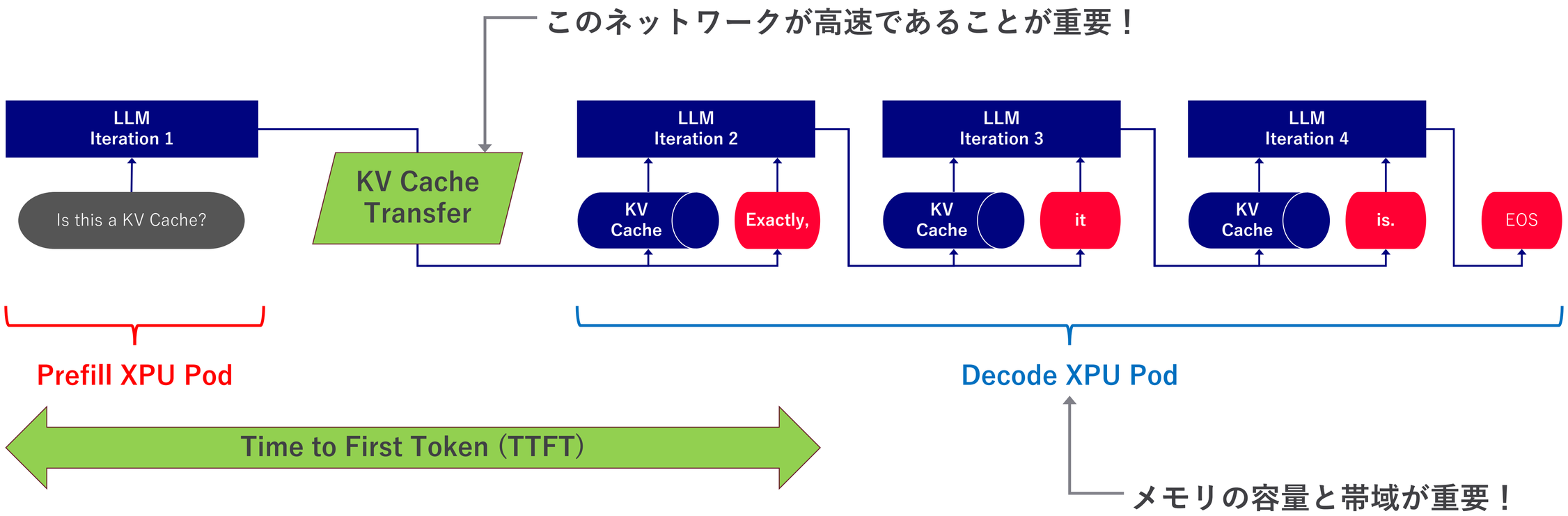
KV Cache Size Calculator を利用して算出
https://lmcache.ai/kv_cache_calculator.html

このKV Cache転送にかかる時間がユーザー体験（TTFT）を悪化させるが、

ユーザーの入力の傾向次第でシステムにかかる負荷が変わるのがインフラ設計（設備投資）上難しいポイント

分散推論においてどのネットワークがいつ使用されるか

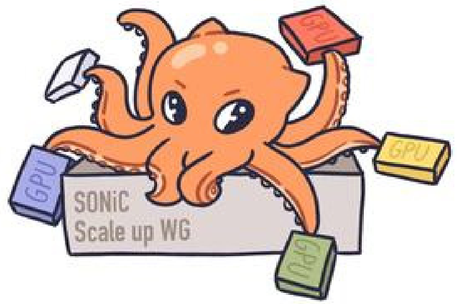
- 「Scale Up」を可能な限り優先して使いたい
  - KV Cacheの転送速度はTTFT（最初の出力トークンを受信するまでの時間）に直結する
  - 分散推論のパフォーマンスチューニングにおいても、高速・低遅延のネットワークを意識する必要がある
  - マルチノードの通信でScale Outを使う場合、そのデータパスの消費電力（Opticsなど）と遅延に影響する



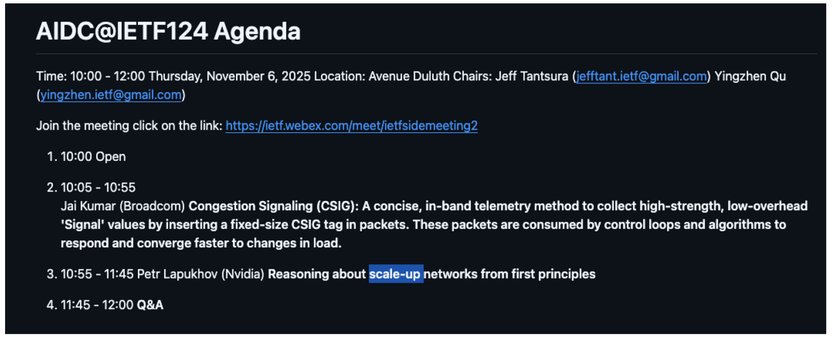
推論基盤におけるScale Upの今後の展望

- 非常に大規模な推論では巨大なメモリプールにKV Cacheを置くことなどが効果的と考えられ、
  分散推論のような構成においてScale Upでの高速・低遅延なメモリアクセスは有効な手段であると認識。
- しかし、（日本国内において）それほど大規模な推論の需要が見込めるかどうかは未知数であると同時に
  現状のScale Upの技術競争はラックスケールやそれと同等規模のXPU間接続システムをスコープとしている。
- 対応するXPUやエコシステムのデファクトが定まっていない現在は、技術動向や議論を注視している状況。

OCP Summit
現地参加・対面議論



SONiC-Scale-Up-WG
https://lists.sonicfoundation.dev/g/SONiC-Scale-Up-WG



AIDC@IETF Side Meeting
https://github.com/Yingzhen-ietf/AIDC-IETF124

まとめ - AIインフラの今後の展望

- XPUチップ単体の性能/メモリ容量には限界があり、モデル規模/要求が伸び続ける以上、
  設計問題の核心は「チップ間接続」に移行・集約されている状況
- これまでの分散学習だけでなく、推論もモデルの大規模化に伴い分散基盤の需要が高まっている
  - ネットワークに超低遅延が求められるユースケースが具体化
- **特にAIインフラにおけるストレージシステムの重要性が増している（筆者の主観）**
  - KV Cacheはもちろん、学習データ、チェックポイントなど用途が拡大

- 選択したデータパスとそのコスト（遅延・消費電力など）を意識することが重要に

組織やビジネスモデル、事業戦略上正しいインフラは
アプリケーションへの正しい理解の上で成り立つ

（参考）分散推論基盤に関する詳説

- 分散推論基盤について、技術の基礎から最新のフレームワークやトレンドまで解説し、
  実際に検証まで行った結果と考察を共有する連載がスタート、全5回程度を予定。

分散推論基盤やその前提の考え方 ～高火力 PHYで作る分散推論基盤 vol.1～
https://knowledge.sakura.ad.jp/48065/

（参考） AIインフラを構築する上での物理面の重要性

- AIインフラはその特有のネットワーク構成や水冷設備の要件などから、複雑・高密度な配線設計やプラントエンジニアリングの領域に突入している
- AIワークロードに加えてすべてのレイヤの知識・経験を導入して構築することが求められている

生成AIインフラを構築してわかったケーブリングの重要性
https://www.janog.gr.jp/meeting/janog56/wp-content/uploads/2025/06/JANOG56-cable-design.pdf

（参考） AIインフラのオブザーバビリティ

- AIインフラはクラウドネイティブ化が進むWebインフラと比較してオブザーバビリティが不足
- AIワークロードの分析などを通してオブザーバビリティ・ギャップを解消する研究開発に取り組む

AIスパコン「さくらONE」の オブザーバビリティ / Observability for AI Supercomputer SAKURAONE
https://speakerdeck.com/yuukit/observability-for-ai-supercomputer-sakuraone