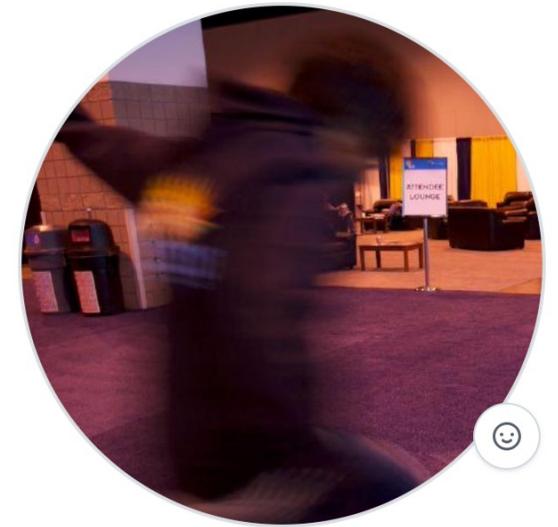


100人超でShowNetの構築にAIを使ってみてわかったこと (と、わからなくなったこと)

東京大学 情報基盤センター
Interop Tokyo 2025 ShowNet NOCチーム
中村 遼

> whoami

- <https://haeena.dev/upa>
 - 東京大学 情報基盤センター
ネットワーク研究部門 准教授
 - 東京大学キャンパスネットワークの運用
 - 研究分野: Networking and Systems
 - Network Architecture, Routing, Protocols, Operation
 - OS Network Stack, Optimization, Virtualization
 - Interop Tokyo ShowNet NOCチーム
 - 主にBackbone + 実験やSDNなどのデモ系



Ryo Nakamura

upa

Edit profile

🔗 156 followers · 14 following

🏠 University of Tokyo

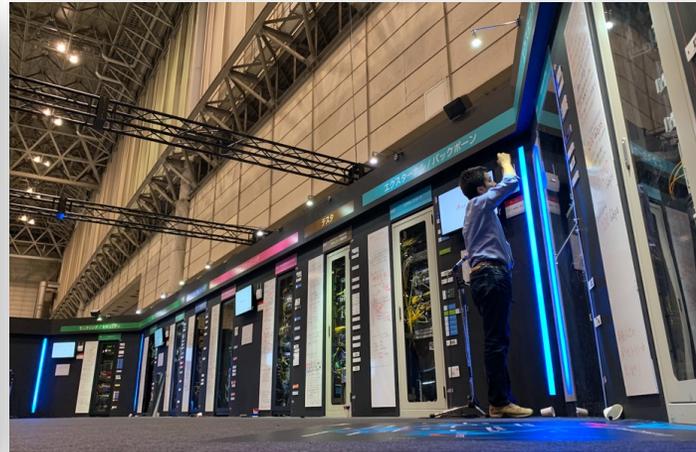
📍 Japan

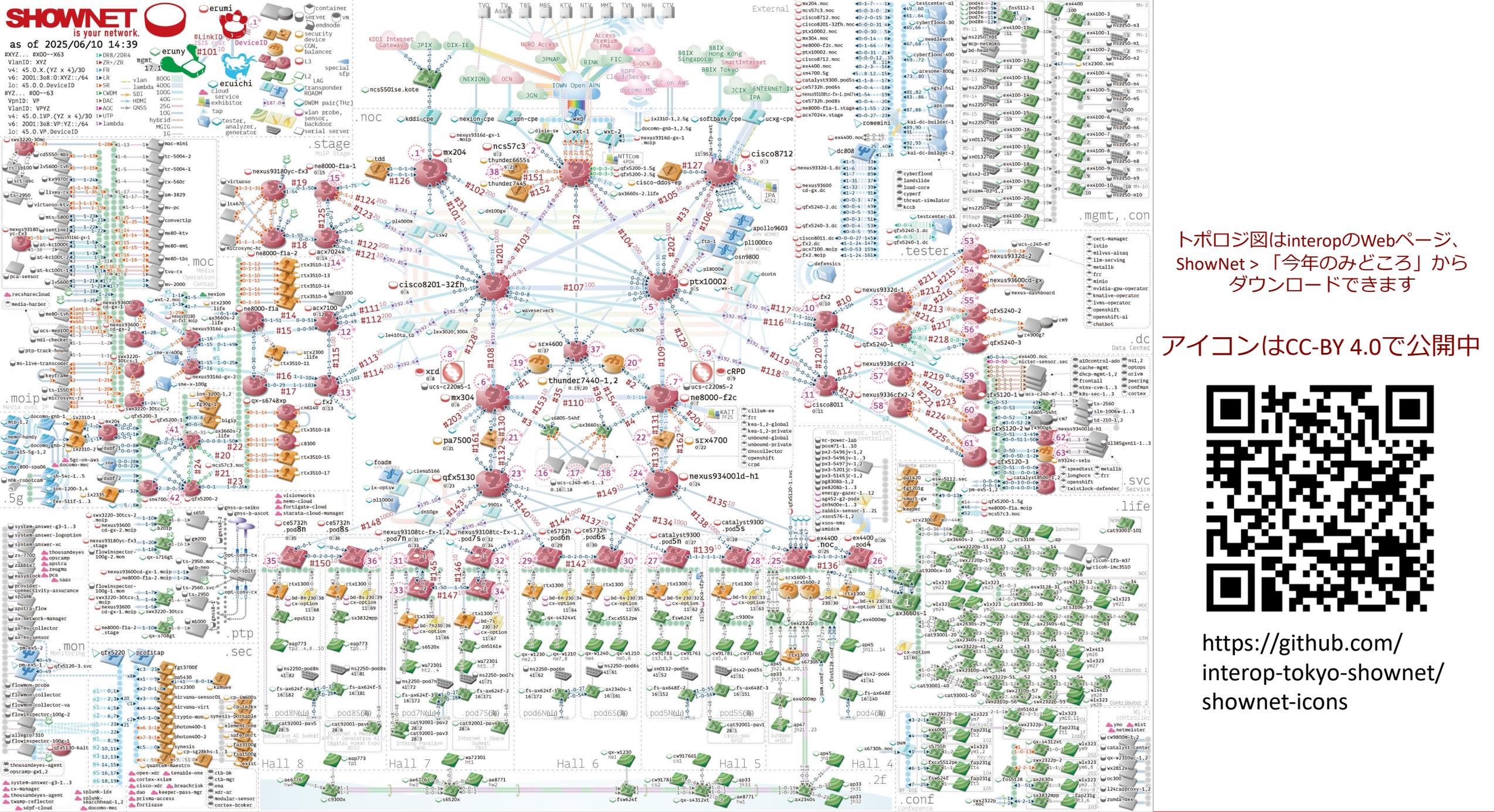
✉ upa@haeena.net

🔗 <https://haeena.dev/upa/>

Interop Tokyo ShowNet

- **Interop:** 世界最大のネットワーク機器と技術の展示会
 - 毎年6月に幕張メッセで開催、2025年は来場者約13万人
- **ShowNet:** 世界最大のデモンストレーションネットワーク
 - その年々の最新技術/プロトコルの相互接続性検証やデモを実施
 - 出展社や来場者へのネットワーク接続性を提供するLive Network





トポロジ図はinteropのWebページ、
 ShowNet > 「今年のみどころ」から
 ダウンロードできます

アイコンはCC-BY 4.0で公開中



<https://github.com/interop-tokyo-shownet/shownet-icons>

生成AIの活用

- Developer/Engineer目線
 - Vibe Coding (Agentic Coding?)
 - コードの生成は自動化
 - 人間は方針を示し、出てきたcodeをReviewする
 - (生産性が上がった下がったの議論は割愛)

これは運用にも使えるはず...!!

障害検知、調査、修正、
チケット起票、クローズ... (広がる夢)

● Read(test/test_e2e.py)
└ Read 751 lines (ctrl+r to expand)

● テストコードのリファクタリングプラン

テストコードを分析した結果、以下のリファクタリング案を提案します：

1. 重複コードの削減

- run2ok/run2ngの使用パターンで繰り返し処理を抽出
- ファイル作成・削除のクリーンアップロジックを共通化

2. テストヘルパー関数の作成

- ファイルペア (src/dst) の作成とクリーンアップを行うヘルパー
- パラメータ化テスト用のファクトリー関数

3. テストデータ管理の改善

- ハードコードされたプレフィックスやサイズ値を定数化
- テストデータ生成の共通化

4. ファイル構成の改善

- test_e2e.py (751行) を機能別に分割
- テーマ別テストファイル (usage, copy, compression, config等)

5. Fileクラスの改善

- util.py:37 のtypo修正 (invalud → invalid)
- コンテキストマネージャー対応でクリーンアップ自動化

6. テストの可読性向上

- 長いテスト関数の分割
- パラメータ化テストの意味を明確にするための命名改善

どの部分から始めますか？全体の実装を進めますか？

> █

? for shortcuts

Days before OpenAI



Days after OpenAI



ProgrammerHumor.io

<https://programmerhumor.io/debugging-memes/how-openai-chatgpt-helps-software-development/>

生成AIにはどれくらいのことができるのか？

- AIの操作によって致命的なトラブルが起きても困る
 - たとえばの極端な話として、、、

なんか通信が遅いです。直して！

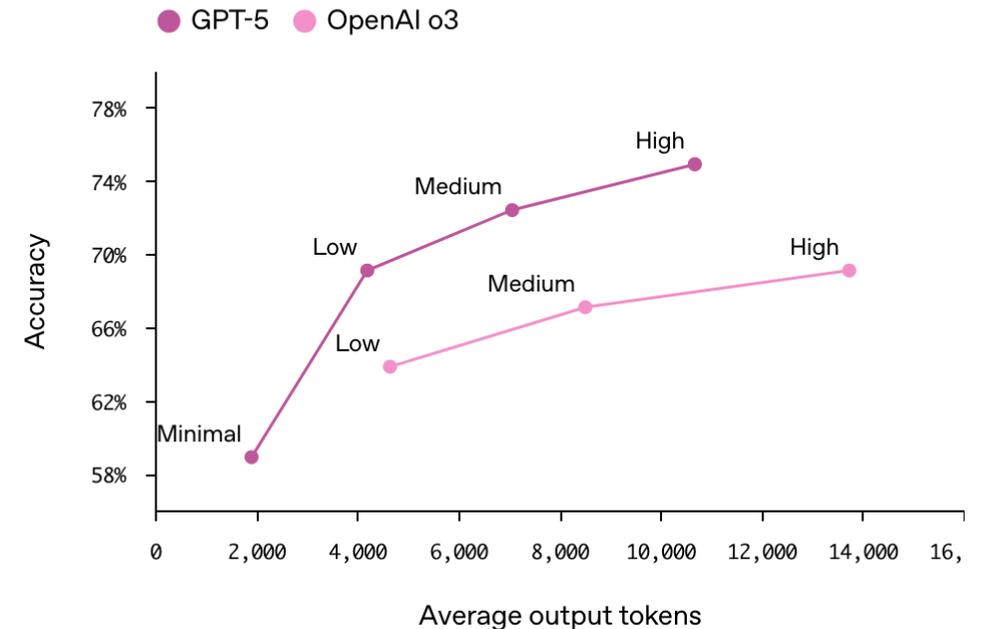


経路中にあるルータを全て再起動します



生成AIはどのようなタスクをどの程度有効にこなせるのか？

SWE-bench Verified
Software engineering

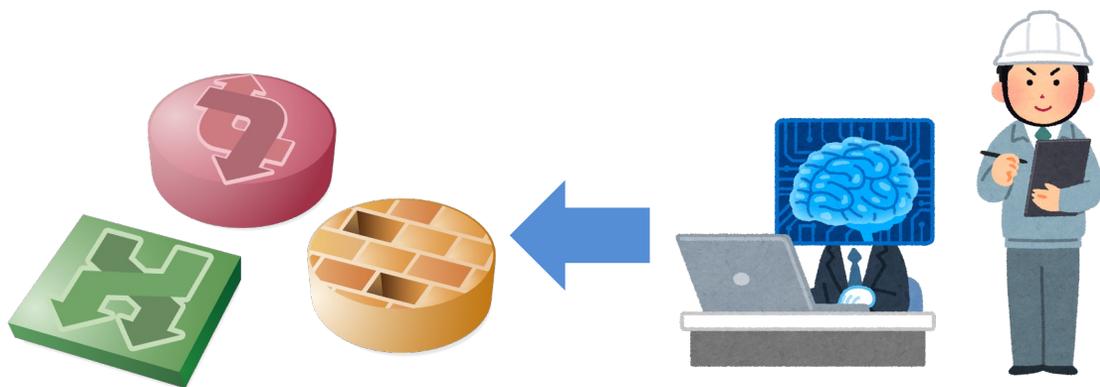


<https://openai.com/ja-JP/index/introducing-gpt-5-for-developers/>

約12日間のべ828名で行うShowNetの構築

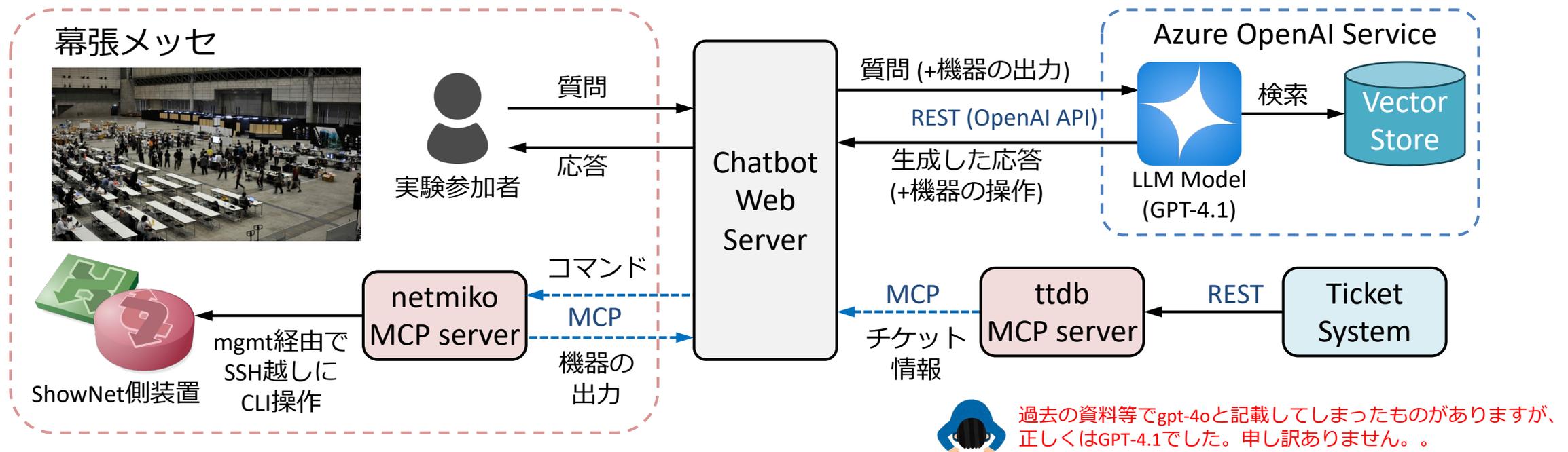
- 生成AIの有効性をベンチマークするのに最高の環境
 - 約2300の機器/製品/サービスから成る超複雑なネットワーク
 - 様々なバックグラウンドを持つネットワークエンジニア
 - NOCメンバー、コントリビュータ、ボランティアのSTM

ShowNetに構築・運用にどれほどAIが使えるのか
100名を超えるネットワークエンジニアが現場で実際に検証



ShowNet構築支援用生成AIチャットロボット

- ネットワークの設計資料などを必要に応じて検索・参照 (RAG)
- LLMがネットワーク機器のCLIにアクセスして直接操作する
- LLMがチケットシステムにアクセスしチケットを参照する



Retrieval Augmented Generation (RAG)

- 生成AIが必要に応じてベクトルストアを検索
- 取得したデータ(ファイルの内容)を応答の生成に利用
- ベクトルストアに入れたファイル
 - 去年のShowNetの機器166台のConfig
 - 今年 ShowNet の設計資料の一部
 - 計8427行のMarkdown (pptx to markdown)
 - 今年 ShowNet の運用ガイド
 - 計1350行のMarkdown

ルータのインターフェースに設定するdescriptionはどうすればいいの？

- インターフェースの `description` 設定ガイドラインは以下の通りです:

基本ルール

1. すべてのインターフェースに設定する:

- 対向の装置およびインターフェースを明記する必要があります。

2. 適用対象範囲:

- バックボーンの物理ポートの他、VLANインターフェース (SVIまたはサブインターフェース) にも設定します。

3. 命名規則に準拠:

- 詳細は対応する命名則に基づきます【12:0+source】。

記述例

- 物理ポートの場合: 対向機器のインターフェース名を記述

```
plaintext
```

```
interface TenGigE0/0/0/0
  description xg-3-1.crs-3.noc
```

- VLANインターフェースの場合:

```
plaintext
```

```
# VLAN
```

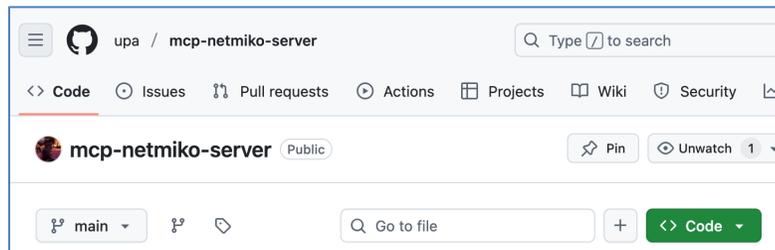
|メッセージを入力してください...



生成AIによるネットワーク機器のCLI操作

最終的には累計29機種、69台のルータ・スイッチがAIから1回以上操作されました

- プロンプトに応じてAIがリアルタイムに機器のCLIを操作する
 - 機器に対して発行できるコマンドを事前に与えているわけではない
- AIと機器のCLIを中継するMCPサーバを実装
 - <https://github.com/upa/mcp-netmiko-server>



ex4100-1.mgmtのroutingの状態を確認、おかしい点があれば報告

- 使用済み call_tool ▾
- ルーティングの状態を確認した結果、おかしい点はありませんでしたが、以下に要点をまとめます。

ルーティング詳細

- **Autonomous System (AS)番号:** 65528
- **Router ID:** 10.16.253.13

メモリのマーク :

- **RIB (Routing Information Base)**
 - ユニークデスティネーション: 最多1407 (2025-05-31)
 - 総ルート数: 最多1433 (2025-05-31)
- **FIB (Forwarding Information Base)**
 - 総ルート数: 最多211 (2025-05-31)



メッセージを入力してください...



例: 機器のシャーシの状態を確認してみる

例: インターフェースの状態を確認してみる

例: 過去のShowNetの機器のconfigから設定例を聞いてみる

例: TTDBのチケットを参照してみる

チャットボットの動作

動画では生成AIがリアルタイムに

- ネットワーク機器のCLIを操作
- チケットを参照しています

On Youtube:

https://www.youtube.com/watch?v=IAa_sjzrdHc

応答の評価

- 実験参加者は任意にチャットを利用し生成AIの応答に👍 と👎 をつける
 - 応答に対する評価の付与自体はベストエフォート
- この評価をもとに、生成AIがどう使われ、どう役に立ったかを分析

🔴 ShowNet 2025のトポロジーによると、**acx7024x.stage**が接続しているルータは以下の通りです：

1. **ptx10002.noc** (100Gリンク)
2. **cisco8201-32fh.noc** (100Gリンク)
3. **ne8000-f2c.stage** (100Gリンク)

これが現在の接続状況の概要です。他に確認したい内容があればお知らせください！



評価結果集計(最終版)

- 評価がついたResponseのうち👍の割合は**68.1%** でした
 - 現時点のLLMをネットワーク運用にそのまま用いる場合のひとつの指標
 - ここからさまざまな向上が考えられる e.g., reasoning, .* engineering

参加者属性	Prompt/Response の総数	👍 Goodの数	👎 Badの数	Goodの割合
All	4007	265	124	68.1%
NOC	673	89	36	71.2%
STM	2798	162	70	69.8%
Contributor	536	14	18	43.8%

ユースケース: Configの確認と修正

- 現在のConfigをプロンプトに入れて間違いを発見・修正
 - この例では正解を提示できた (maximum-pathsとmultipath-relax)



nexusでeBGPでmultipathしたいんだけど、
下記のconfigだと経路が一つしか入りません。
何かmultipathを有効化するconfigが必要？

```
router bgp 65353
router-id 45.0.0.53
log-neighbor-changes
address-family ipv4 unicast
  redistribute direct route-map redist-connected
neighbor 2001:3e8:0:211::51
  remote-as 65351
  description nexus9332d-1.dc
...(以下略)
```

NexusシリーズでeBGPでMultipathを有効化するには、
以下の設定が考えられます。

1. `maximum-paths`コマンド使用: `router bgp`の下で、
使用可能な複数パスの数を指定する`maximum-paths`
設定が必要です。例えば、次のように配置します。

```
router bgp 65353
address-family ipv4 unicast
  maximum-paths 64
address-family ipv6 unicast
  maximum-paths 64
```

```
router bgp 65353
  bestpath as-path multipath-relax
```



ユースケース: 機器の状態確認 + 反復実行

- ある箇所でログを確認、同じ操作を別の機種でも反復実行

pod5でstorm-controlのsyslogがでているか確認

`pod5n`スイッチでストームコントロールに関するsyslogが出力されていることが確認されました。その内容は以下の通りです：

```
Jun 12 09:22:49.742: %STORM_CONTROL-3-TRAP: A packet storm was detected on Multicast. Sending SNMP trap.  
Jun 12 09:22:49.742: %PM-4-ERR_DISABLE: storm-control error detected on Tw1/0/22, putting Tw1/0/22 in err-disable state  
Jun 12 09:22:49.745: %STORM_CONTROL-3-SHUTDOWN: A packet storm was detected on Tw1/0/22. The interface has been disabled.
```

一方で、`pod5s`スイッチではストームコントロールに関連するsyslogは出力されていませんでした。

同じようにpod4,pod6,pod7,pod8でもsyslogを確認して

コマンドの間違い

- コマンドを間違えることはやはりある
 - 同様にConfigが意図しないものになることも

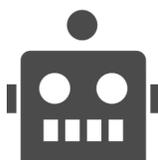
コアラータのISISが設定してあるリンクでMTUが9000になっていないリンクを探してください



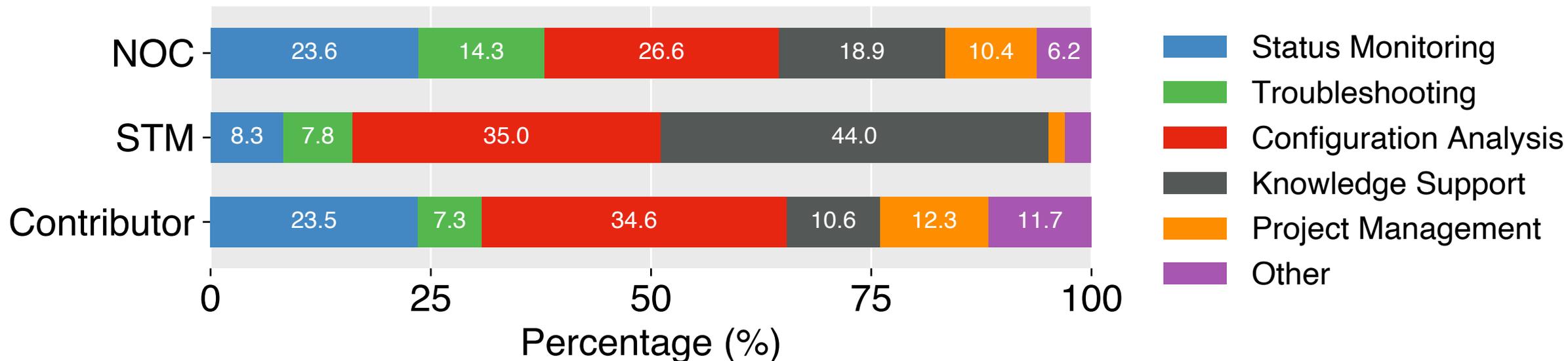
コアラータについてISISリンクのMTU設定を確認するコマンドを実行しましたが、コマンドが正しく機能しないデバイスがあるようです：

1. Juniper (mx204.noc, ptx10002.noc) : ``show isis interface | match inet | match MTU``を実行しましたが、明確な出力が得られませんでした。
2. Cisco IOS-XR (ncs57c3.noc, cisco8712.noc, cisco8201-32fh.noc) : 使用されたコマンド ``show clns interface brief | include MTU`` が無効であると認識されました。

これにより、エラーが発生した理由を確認し、別のアプローチまたは適切なコマンドを検討する必要があります。設定変更が必要な場合や詳細確認が必要であればお知らせください。



ユースケースの分類と内訳



- 対象を理解している方がより直接的に運用にAIを活用
 - NOCは状態確認とトラシューが37.9%
- 生成AIを教師的/知識補助的に使うのも有効
 - STMはConfigの分析と知識補助の利用が全体の79%

AIを使う側のスキルセットや背景知識によりAIの使い方も変わってくる

ユースケースごとのPositive Rate

- 調査・知識系はGood
 - Status Monitoring
 - Configuration Analysis
 - Knowledge Support
- トラッシュューは厳しめ
 - 57.6% と半分ちょい
 - Reasoning Modelになればあるいは？

カテゴリー	👍の割合
Status Monitoring	77%
Trouble Shooting	58.7%
Configuration Analysis	68.9%
Knowledge Support	70.2%
Project Management	50%
Other	100%

詳細は論文(これから公開)をご参照ください

- IEEE Network Operations and Management Symposium (NOMS) 2026
 - arxivにあげようとしています(endorsement待ち...)

How Helpful is LLM Assistance in Network Operations? A Case Study at a Large Demonstration Network

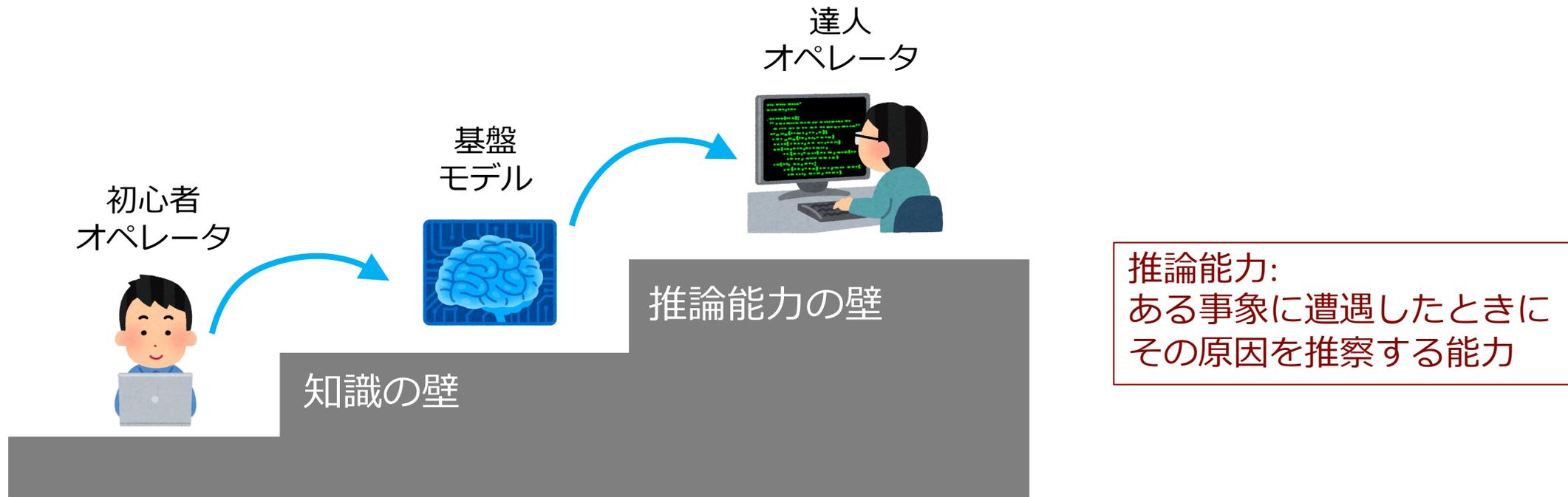
Ryo Nakamura
Information Technology Center
The University of Tokyo
Tokyo, Japan
upa@nc.u-tokyo.ac.jp

Koshi Eguchi*
Graduate School of Information Science and Technology
The University of Tokyo
Tokyo, Japan
koshiegu@gmail.com

実験と分析を終えて

ネットワーク運用の観点から見た 2025年5月時点の生成AIの現在地

- 現在の基盤モデルの感触は「初心者以上、達人未満」
 - 知識とスループットは間違いなく人間よりAIの方が上
 - トラシューにおける推論能力は【まだ】熟練したオペレータの方が上



大規模言語モデル自体の進歩

2025/9/29登場

- claude-sonnet-4.5でNETCONの問題を30/33は自力で解ける
 - 2025年のShowNetの実験ではReasoning Modelではなかった
- LLMの性能が上がり続けるなら、本当になんでも解けるかも?



NETCONをLLMで解く
-JANOG57 Day1-

金井 瑛 (株式会社Preferred Networks)
久保京介 (NTTドコモビジネス株式会社)



```
gx10-1 ~ > ollama run gpt-oss:120b
>>> OSPFとIS-ISの違いってなんだろう？
Thinking...
The user asks in Japanese: "OSPFとIS-ISの違いってなんだろう？" They want differences
between OSPF and IS-IS. Should answer in Japanese, explain differences: protocol type,
design, link-state, hierarchical, metric, area, authentication, support for IPv6,
scalability, convergence, vendor support, implementations, etc. Provide bullet points,
maybe table. Also mention use cases. Probably concise but thorough. Should be in
Japanese.
...done thinking.

## OSPF と IS-IS の主な違い

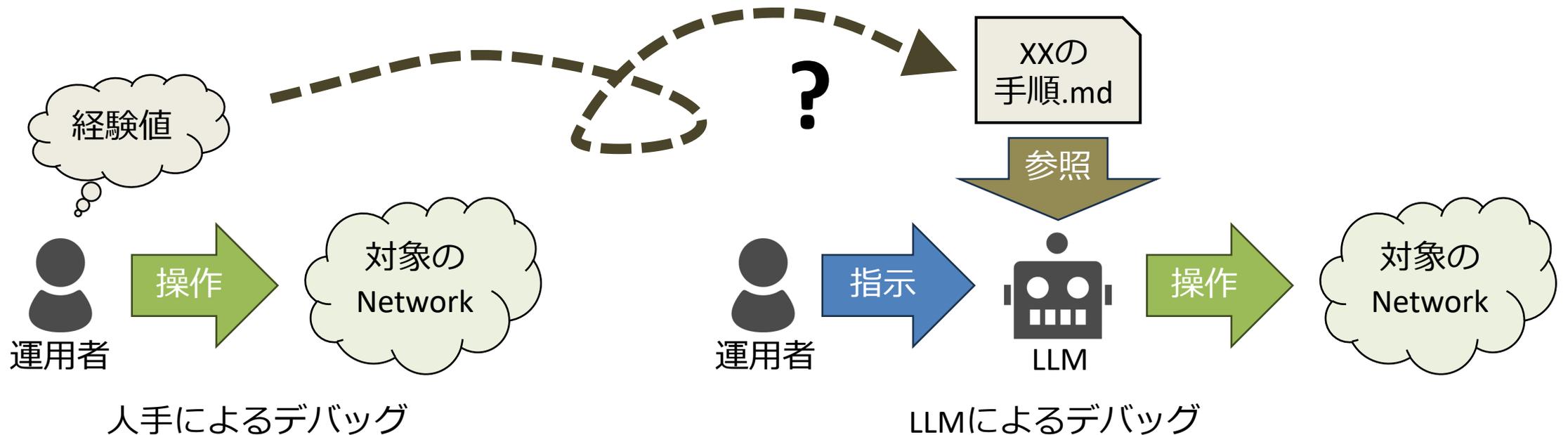
| 項目 | OSPF (Open Shortest Path First) | IS-IS (Intermediate System-Intermediate
System) |
|-----|-----|-----|
| **標準化団体** | IETF (RFC 2328 / RFC 5340) | ISO/IEC (ISO 10589) |
| **プロトコル層** | IP 上で動作 (IP-based) | データリンク層 (OSI 第2層) で動作し、IP
とは独立 |
| **アドレス族** | IPv4 と IPv6 (Separate プロトコル : OSPFv2 / OSPFv3) | IPv4 と IPv6
```

出典: NETCONをLLMで解く -JANOG57 Day1-

gpt-oss:120b running on ollama

LLMに与える指示と手順

- LLMに対して自然言語で指示を与える
- 指示が達成可能であれば、LLMはその指示を完遂できるだろう
 - 達成する上で必要になる外部との作用 (MCP, shell) と、手順 (SKILL) があれば
- 自分のトラシューに関する経験知をSKILL.mdに落とし込もうとしたが、
 - 自分が言語化が下手なだけの可能性も



初心者エンジニアが生成AIを超えるには?

(または初心者(に限らず?)エンジニアがこの先生きのこるには)

• 現時点では明確な答えは無い...

- ネットワーク以外のあらゆる領域でも同様に課題
 - AIにやらせておくだけでは、人間の腕は上がらない
 - 一方、別の腕が上がるという説はあると思う (ASM→Fortranの変化のように)
- これからプログラミング/ネットワークを学び始める人は、どのタイミングでAIの出力をValidateできるようになるのだろうか？
またはAIの出力をValidateする必要もなくなるのか？



おわりに

- ネットワーク運用と生成AIと
 - 2025末時点でもうすでにかなりイケる
 - 環境に合わせた手順/指示は必要。求められる言語化能力
 - 全て生成AIにやらせればよくなったら、これから学び始めるエンジニアは一体どうすればいいのだろうか

• Interop Tokyo ShowNet

- 未来のネットワークのひとつのカタチ
- 実験・検証とコミュニティへの還元
- 2026年は6/10 ~ 6/12 @幕張メッセ

