

ナビタイムが導く AWS活用への最適経路

株式会社ナビタイムジャパン

2018.06.01

アジェンダ

- ▶ ナビタイムジャパンの紹介
- ▶ ナビタイムジャパンのAWS活用
 - ▶ データ分析基盤
 - ▶ ナビタイムサービス
 - ▶ GPUでの超並列経路探索の取組み
- ▶ まとめ

自己紹介

田中 一樹

株式会社ナビタイムジャパン

インフラエンジニア クラウド担当



業務

AWS問わずクラウド周りのアーキテクト、構築などなど

NAVITIME

会社紹介

会社名

株式会社ナビタイムジャパン

設立

2000年3月1日

業務内容

ナビゲーションサイト・アプリの運営・開発
通信カーナビゲーション事業
経路探索エンジンのライセンス事業 など

従業員数

約440名（エンジニア80%以上）

会社紹介

公共交通



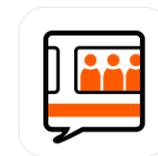
NAVITIME



乗換
NAVITIME



バス
NAVITIME



こみれぽ

ドライブ



ドライブ
サポーター



カーナビタイム



トラックカーナビ

ツーリング



ツーリング
サポーター



自転車
NAVITIME

海外



NAVITIME
for Japan
Travel



NAVITIME
Transit

トラベル



NAVITIME
Travel



Plat by
NAVITIME

ヘルスケア



ウォーキング
NAVITIME

PC/SPブラウザ



会社紹介



月間ユーザ数

約**4100**万UU

有料会員数

約**480**万人

(2017年12月末時点)

会社紹介

ビジネスナビタイム
交通費精算パッケージ
交通費精算システムの決定版



ビジネスナビタイム
動態管理ソリューション
スマートフォンを活用した新しい動態管理



広告 / アライアンス

3000万人が利用するNAVITIMEで人と企業をつなぐ



オウンドメディア

自社メディアの価値向上・サービス提供支援



交通コンサルティング

ビッグデータを用いた分析・提案



電鉄・バス事業者向けソリューション

高品質の乗換・時刻表検索、その他付加価値のあるコンテンツの提案・提供



API / SDK

経路探索エンジン、ナビエンジンのご提供



テレマティクスサービス

車載向けサービスの提供



ナビタイムジャパンのAWS活用

データ分析基盤

NAVITIME

AWS 導入事例

株式会社ナビタイムジャパン

Amazon Athena を利用することでインフラコストを 75% 削減

“ AWS 採用を掲げクラウド化を前面に打ち出したところ、インフラコストが削減されました。また、エンジニアにコスト意識が芽生えました。 ”
株式会社ナビタイムジャパン

月間ユーザー数約 4,100 万のナビゲーションサービス

コア技術の経路探索エンジンをベースに、ナビゲーションに関するさまざまなサービスを提供している株式会社ナビタイムジャパンは、『自転車 NAVITIME』や『ドライブサポーター』など、個人向けサービスを各種提供し、トラベル事業に参入。インバウンド需要の拡大を受けて多言語対応するなど、『動態管理ソリューション』、お客様サイト内で店舗までのルートを案内する『NA

サーバーレスの Athena の採用によりすべての課題を解決し、インフラコストを 75% 削減

解決策を模索する中、AWS re:Invent 2016 においてサーバーレスのクエリサービス Amazon Athena がリリースされる情報を入手したナビタイムジャパンは、採用を即断しました。インフラエンジニアの新立和広氏はその理由を次のように話します。「ログデータの転送コストやセキュリティの課題を一気に解決できることが決め手となりました。また、マネージドサービスであるため運用負荷が軽減できることや S3 を単一のデータレイクとして集約し、シンプルな管理ができること、使い慣れたかつ実績のある AWS のサービスであることも評価しました。」

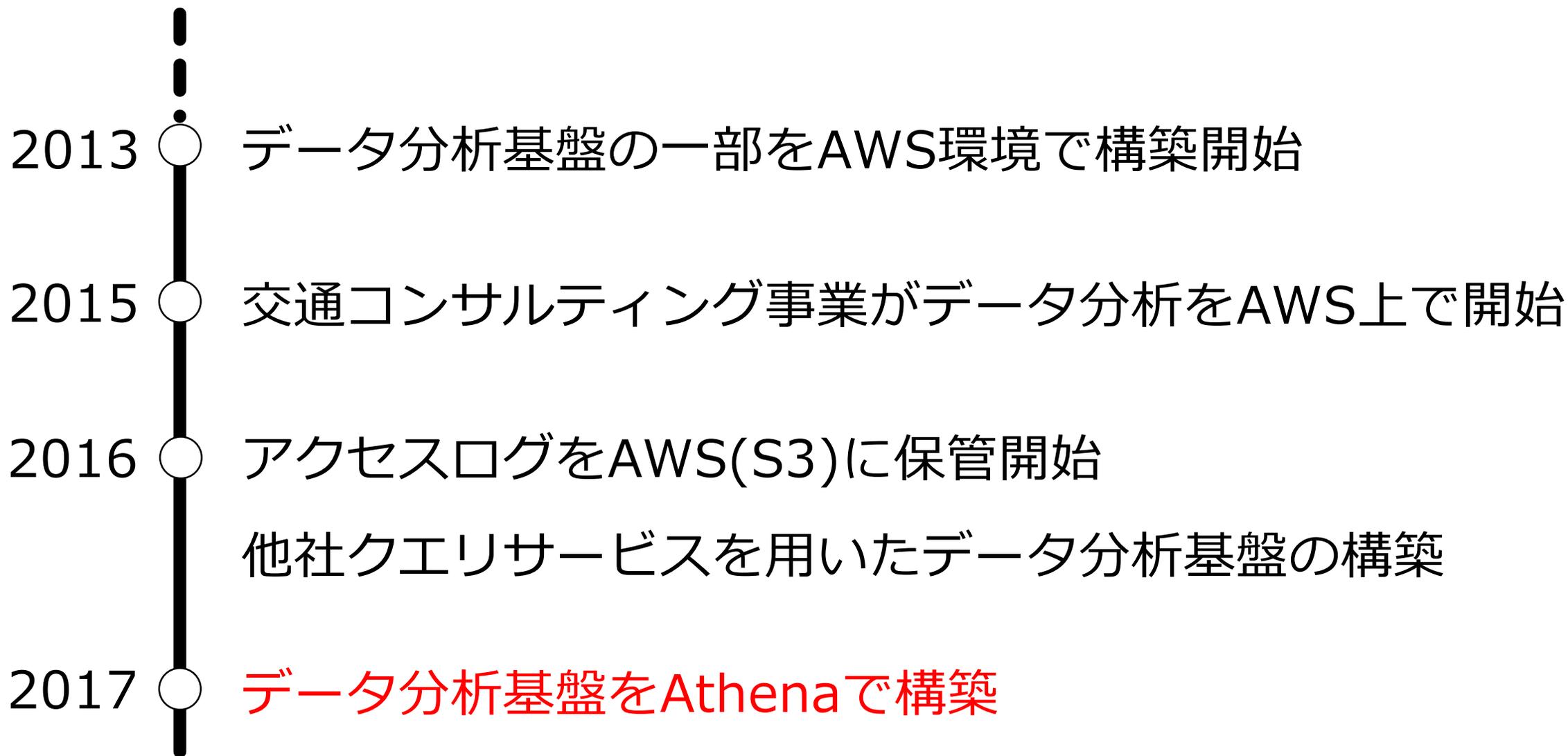
現在は、大量のログデータを EMR で匿名化処理を施してから S3 に保存し、これらのデータに対して Athena で分析しています。「S3 と Athena が同一基盤となることでデータ転送と転送先のストレージが不要となり、インフラコストは全体で 75% を削減できました。1 日に約 8 時間かけていたデータ転送がなくなったことで、2 日後でなければ見られなかったログ情報が翌日に確認できるようになりました。詳細な権限管理も実現しています。」(田中氏)

Athena を採用することによって、法人向けの交通分析サービス「道路プロファイラー」をリリースできたことも大きな成果です。道路プロファイラーとは、『カーナビタイム』や『NAVITIME ドライブサポーター』などのカーナビアプリから取得した携帯カーナビプローブデータとアプリ利用者の属性情報を組み合わせ、全国の道路を対象に経路分析や渋滞分析、走行車両の属性分布を集計・可視化することができるサービスです。「分析レスポンスが求められるこのサービスを提供するにあたり、さまざまなものを試してみましたが、Athena 以外に要求を満たすものはありませんでした。サーバーレスでスケーラブルな Athena でなければ実現できなかったサービスだと言えます。」と新立氏は語ります。

導入効果はこれだけにとどまりません。インフラ部門における人材確保の効果も生まれています。「オンプレミス時代のインフラ部門はサーバーを 24 時間見守るイメージが強く、配属希望がほとんどありませんでした。AWS 採用を掲げクラウド化を前面に打ち出したところ、エンジニアのモチベーションが一気に高まり、インフラ部門に関心を持つエンジニアや配属希望者が急増し、人材確保の悩みが解消されました。」(菊池氏)

加えて、エンジニア自身がコスト意識を身につけたことも見逃せない効果です。「以前は部門の要求に応じてサーバーを随時調達して払い出すだけでしたが、AWS によって毎月のコストが可視化され、コスト意識が身につきました。その結果、経路探索エンジンの開発部門では、サーバーのパフォーマンスを強化したうえで約 20% のコスト削減を実現しています。」(菊池氏)

ナビタイムジャパンのデータ分析変遷

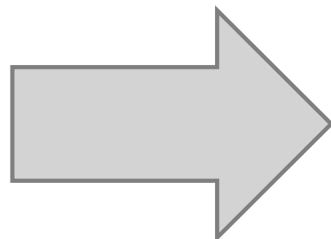
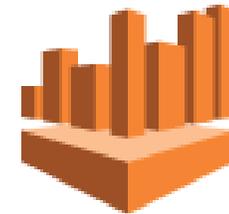


データ分析基盤

- ▶ ナビタイムでのログ分析のポイント
 - ▶ 集計に時間がかからない
 - ▶ 数秒～数分で終わる
 - ▶ 集計にコストをかけない
 - ▶ 安く
 - ▶ 詳細なアクセス管理ができる
 - ▶ アクセスログの閲覧権限は厳しく

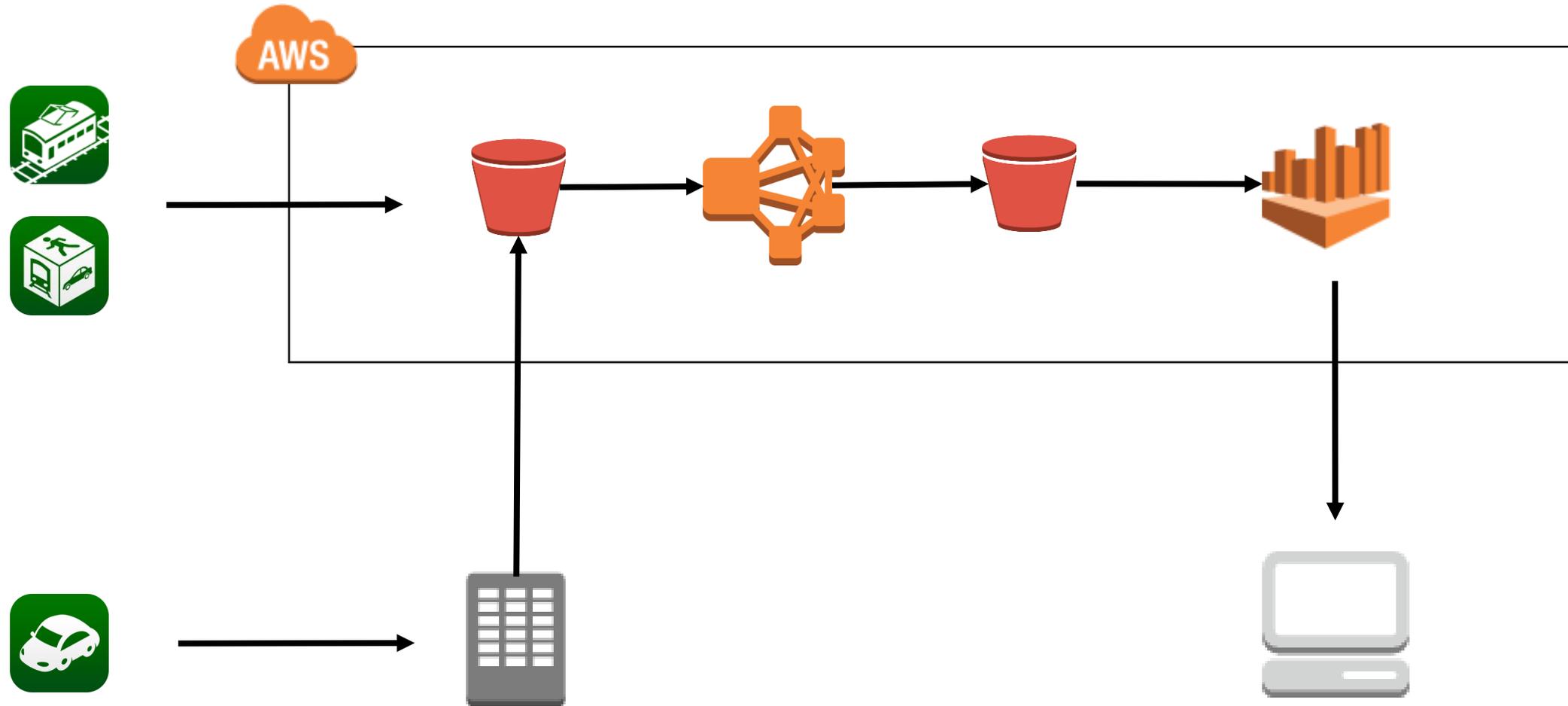
データ分析基盤

- ▶ ナビタイムでのログ分析のポイント
 - ▶ 集計に時間がかからない
 - ▶ 数秒～数分で終わる
 - ▶ 集計にコストをかけない
 - ▶ 安く
 - ▶ 詳細なアクセス管理ができる
 - ▶ アクセスログの閲覧権限は厳しく



Amazon Athenaが最適

データ分析基盤



データ分析基盤

- ▶ データ分析基盤構築にあたり、コンプライアンス/セキュリティチームから出た要望
 - ▶ 各利用者の見られるログは適切に管理してほしい
 - ▶ 誰がどんなクエリをどれくらい発行したか監視したい
 - ▶ 社外からログへのアクセスをさせたくない

データ分析基盤

- ▶ データ分析基盤構築にあたり、コンプライアンス/セキュリティチームから出た要望
 - ▶ 各利用者の見られるログは適切に管理してほしい
 - ▶ 誰がどんなクエリをどれくらい発行したか監視したい
 - ▶ 社外からログへのアクセスをさせたくない

Amazon AthenaとAWSなら**すべて解決**



データ分析基盤

- ▶ 各利用者の見られるログは適切に管理してほしい

データ分析基盤

- ▶ 各利用者の見られるログは適切に管理してほしい
 - ▶ IAMポリシーで参照できるバケットを絞る



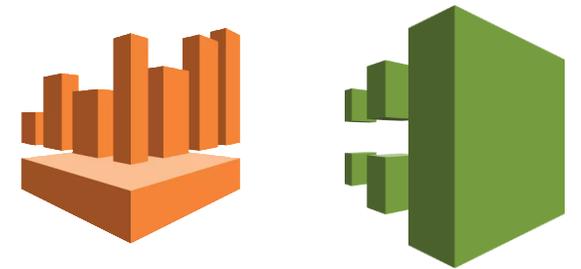
- ▶ 現時点で、DB/テーブル単位での権限絞込はできない
 - ▶ 代わりにS3バケット単位でGetObjectの権限を絞る

データ分析基盤

- ▶ 誰がどんなクエリをどれくらい発行したか監視したい

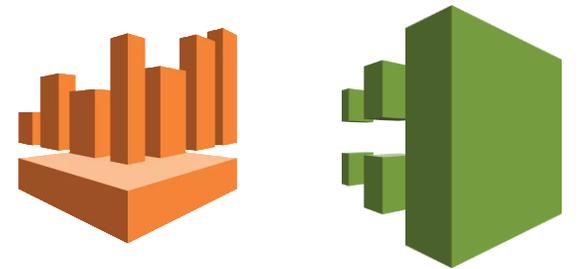
データ分析基盤

- ▶ 誰がどんなクエリをどれくらい発行したか監視したい
 - ▶ Athenaのクエリ履歴 + CloudTrailから追跡
 - ▶ Athena : どのクエリをどれくらいスキャンしたか
 - ▶ CloudTrail : どのユーザーがどのクエリを実行したか



データ分析基盤

- ▶ 誰がどんなクエリをどれくらい発行したか監視したい
 - ▶ Athenaのクエリ履歴 + CloudTrailから追跡
 - ▶ Athena : どのクエリをどれくらいスキャンしたか
 - ▶ CloudTrail : どのユーザーがどのクエリを実行したか
 - ▶ この2つのログをAthenaのテーブルに投入
 - ▶ 突合して、1日1回事後チェック
 - ▶ 1GByte以上スキャンしたユーザー
 - ▶ クエリに不備がありそうなユーザー



にslackで通知

データ分析基盤

- ▶ 社外からログへのアクセスをさせたくない

データ分析基盤

- ▶ 社外からログへのアクセスをさせたくない
 - ▶ IAMポリシーでSourceIPをオフィス限定にする
 - ▶ Athenaの実行時にSourceIPがamazonaws.comになるため、普通に設定するとRunQueryもできなくなる
 - ▶ スイッチロールを使いこの問題を解決

データ分析基盤

▶ スイッチロール

- ▶ IAMロールに割り当てられた権限を一時的にIAMユーザーに割り当てる
- ▶ SwitchRoleの切り替え元はIPやユーザーなどで制限可能



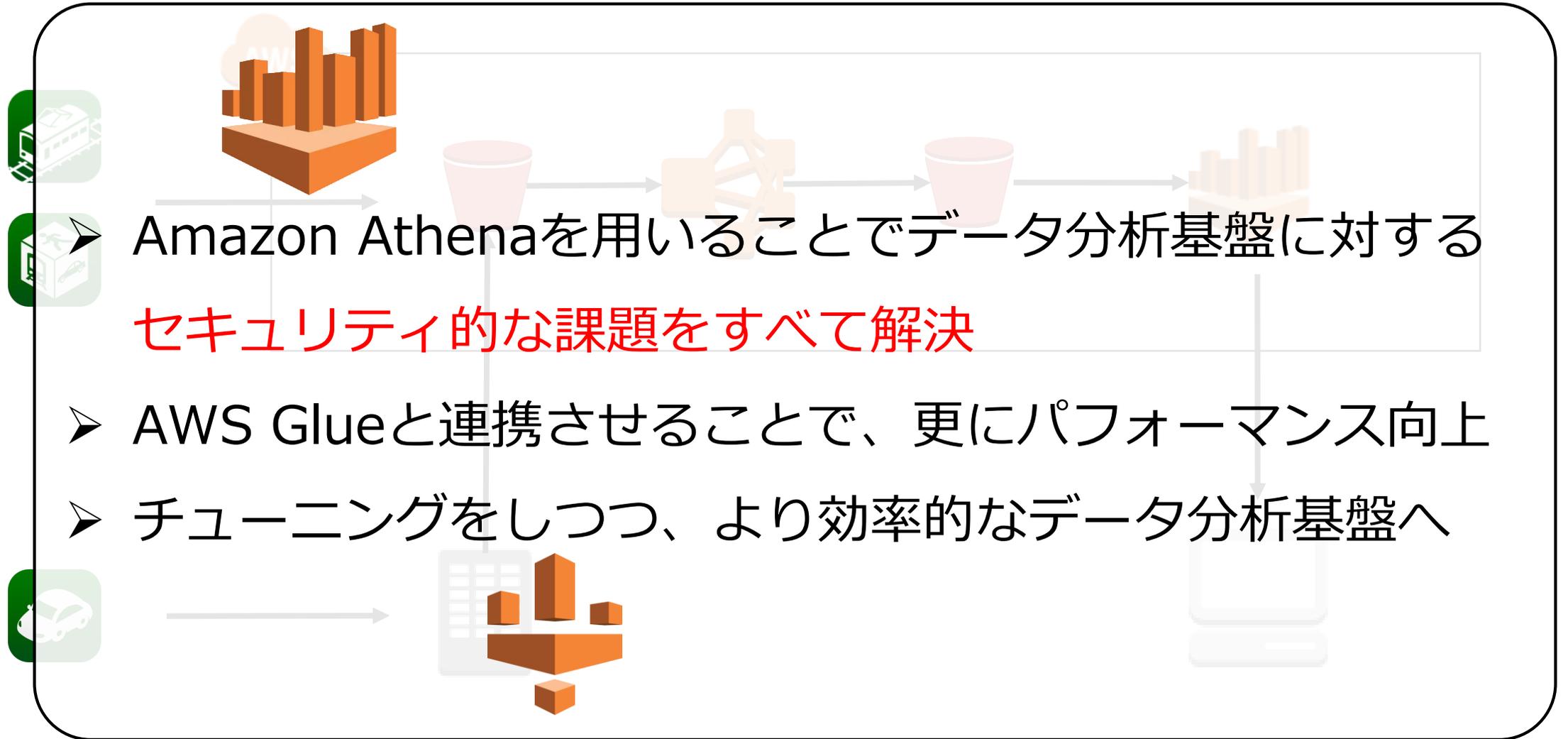
データ分析基盤



- ◆ IPアドレス制限なし
- ◆ MFAの有効化
- ◆ スイッチロールのみ許可

- ◆ スイッチ元のIPアドレス制限(信頼関係)
- ◆ スイッチ元のユーザーを固定(IAMグループ)
- ◆ IAMグループごとにロールを作成
- ◆ Athena/S3/KMSの権限のみ

データ分析基盤



ナビタイムサービスのAWS活用

ナビタイムサービスのAWS活用

NAVITIME TransitをAWS環境で構築

2016

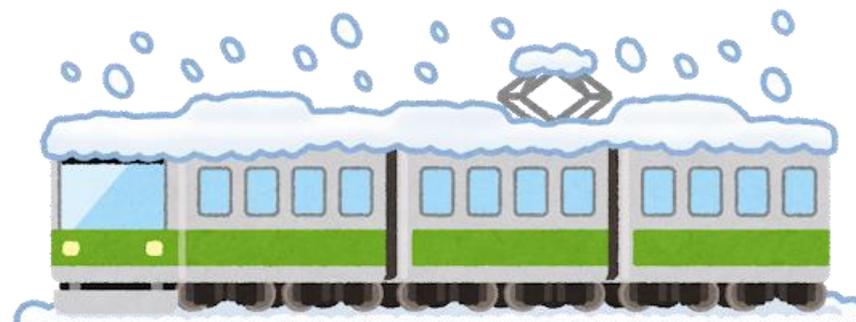


ナビタイムサービスのAWS移行プロジェクトスタート

AWS移行の背景



- ▶ ナビタイムのアクセス推移は**外的要因が多い**
 - ▶ 例) 電車遅延、天候不良、GW
- ▶ 外的要因を見越したサーバ増強は**都度工数がかかる**
- ▶ ユーザーが必要なときに、サービス提供ができなくなる状況をなくすため**AWSへの移行を決意**
- ▶ すでにNAVITIME TransitをAWS上で稼働している実績あり



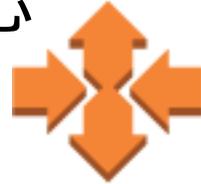
AWS移行の方針

- ▶ アクセス増加時に自動でスケールできるように
- ▶ 構築はすべてCode化
- ▶ アプリケーションのコンテナ化

AWS移行の方針

- ▶ アクセス増加時に自動でスケールできるように

- ▶ Auto Scalingで対応



- ▶ 構築はすべてCode化

- ▶ CloudFormationで構築



- ▶ アプリケーションのコンテナ化

- ▶ ECSで構築



ナビタイムサービスのAWS活用



NAVITIME TransitをAWS環境で構築

2016

ナビタイムサービスのAWS移行プロジェクトスタート

2017

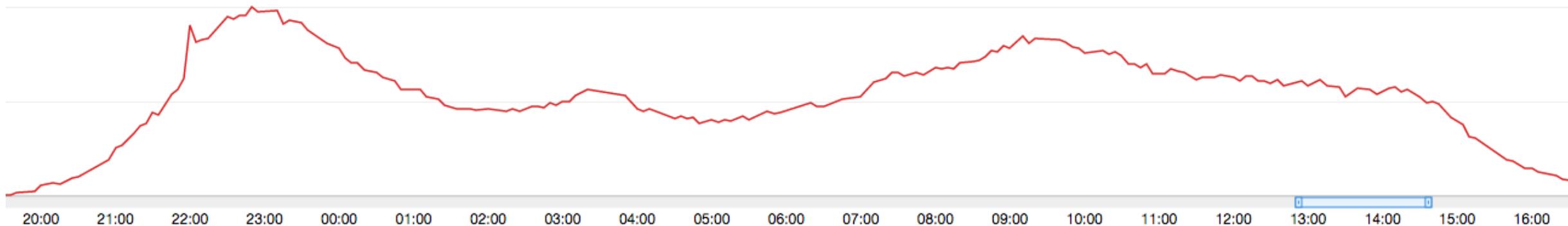
乗換NAVITIMEのAWS移行

乗換NAVITIMEのAWS移行

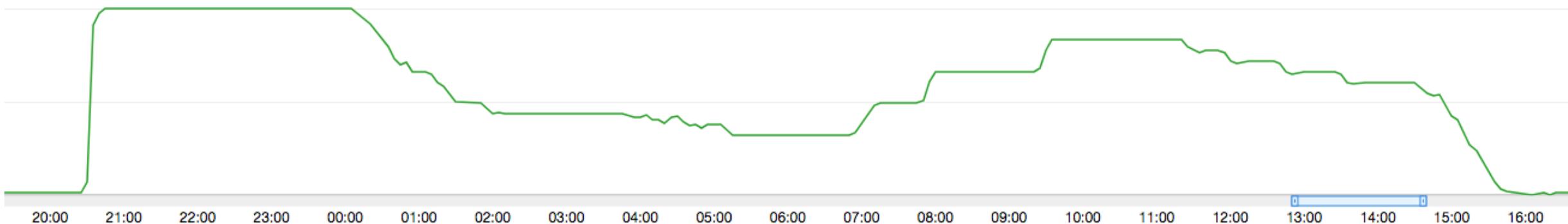
- ▶ 昨年度のAWS Summit Tokyoで紹介
 - ▶ ナビタイムサービスにおける、Amazon ECS を活用したシステム移行～『乗換NAVITIME』での移行事例～
- ▶ 移行が完了し1年以上経過
 - ▶ サーバ輻輳による障害が大幅減
 - ▶ AWS移行直後の利用費増加問題が解消

乗換NAVITIMEのAWS移行

▶ アクセス推移

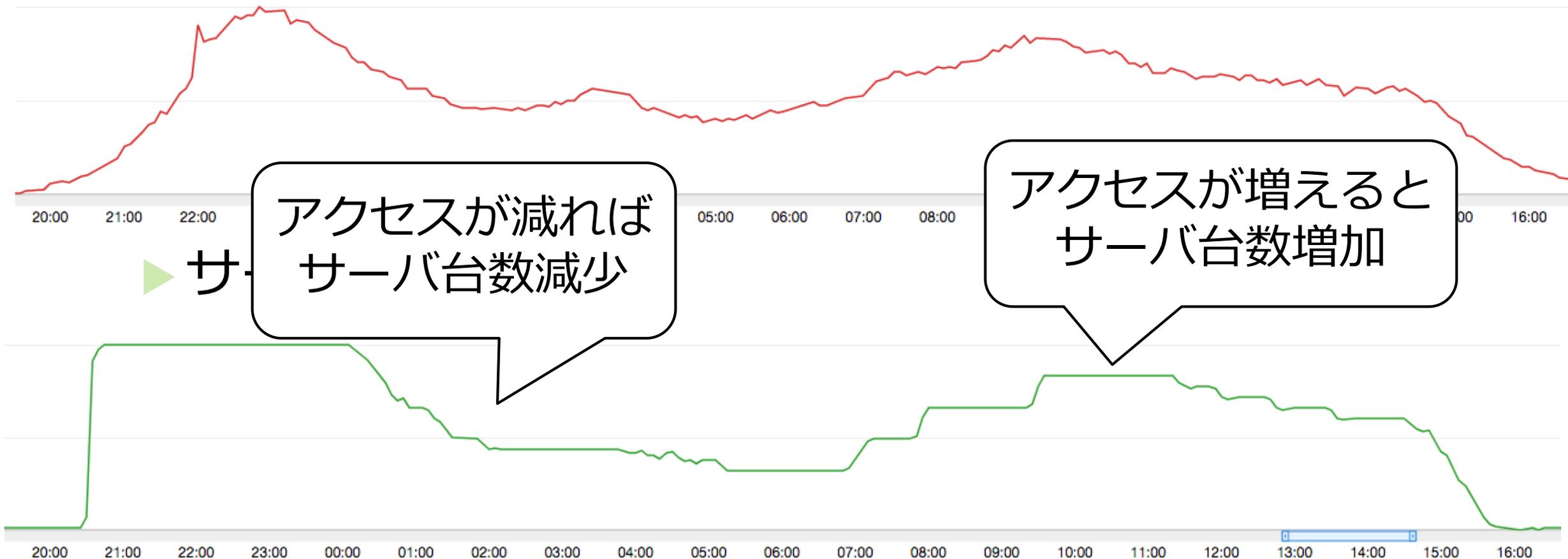


▶ サーバ台数推移



乗換NAVITIMEのAWS移行

▶ アクセス推移



乗換NAVITIMEのAWS移行

▶ AWS利用費

▶ 移行直後

- ▶ EC2/RDS : オンデマンドインスタンス = 費用がかさむ
- ▶ 並行運用期間のオンプレサーバ費も継続

▶ 現在

▶ 費用問題を解消

- ▶ EC2 : スポットフリート&リザーブドインスタンス
- ▶ RDS : リザーブドインスタンス

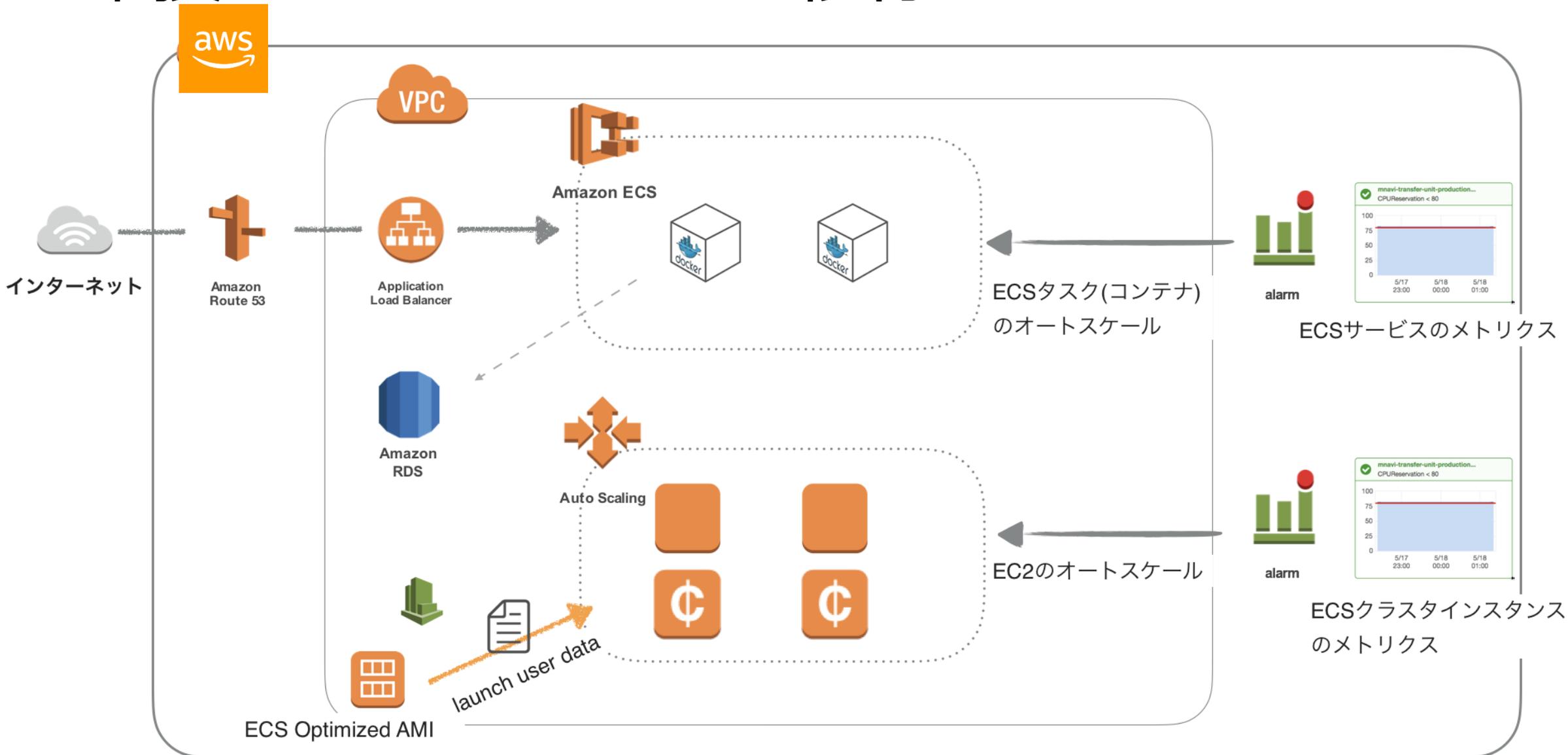
乗換NAVITIMEのAWS移行

- ▶ スポットフリート 
 - ▶ スポットインスタンスの集合
 - ▶ フリート全体でターゲットの容量を満たす
 - ▶ Auto Scalingにも対応
 - ▶ 複数のAZ、複数のインスタンスタイプを指定可能なため、可用性を向上させることが可能

乗換NAVITIMEのAWS移行

- ▶ スポットフリートの利用
 - ▶ コストメリット
 - ▶ オンデマンド料金の最大90%オフ
 - ▶ 相性メリット
 - ▶ スポットインスタンスは突然シャットダウンする
 - ▶ コンテナ化&ドレイニング処理を入れているのでシャットダウンしても問題なし

乗換NAVITIMEのAWS移行



ナビタイムサービスのAWS活用



NAVITIME TransitをAWS環境で構築

2016

ナビタイムサービスのAWS移行プロジェクトスタート

2017

乗換NAVITIMEのAWS移行

ウォーキングNAVITIMEのAWS移行

NAVITIMEのAWS移行

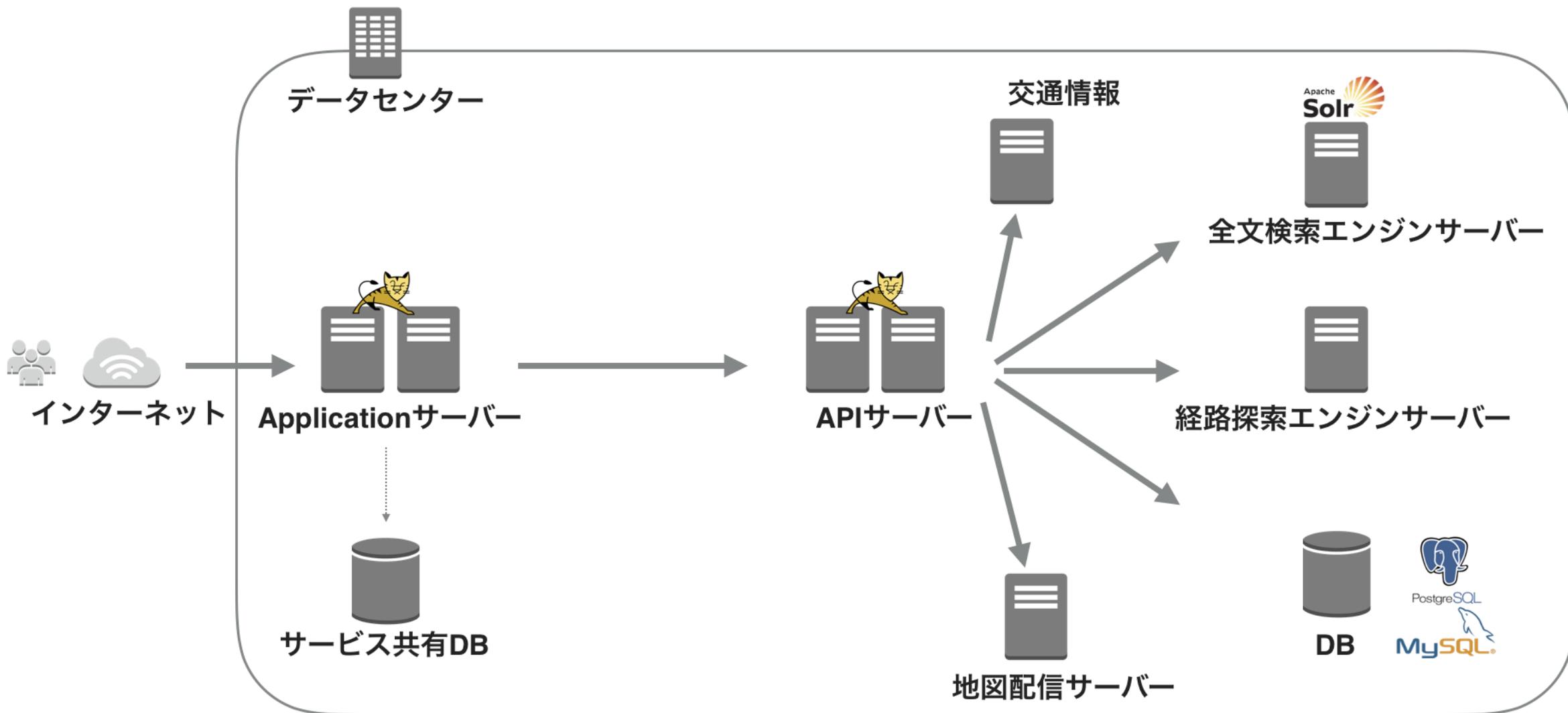
NAVITIMEのAWS移行

- ▶ AWS移行第3弾
- ▶ ナビタイムのメインサービス



- ▶ ドアtoドアの経路探索が可能
- ▶ 公共交通機関+車+徒歩の経路探索が可能
 - ▶ 混雑を避けたルート
 - ▶ 終電後に帰宅可能なルート

NAVITIMEのAWS移行



NAVITIMEのAWS移行

- ▶ 乗換NAVITIMEと異なる点
 - ▶ 地図の表示
 - ▶ 公共交通機関以外の経路探索
 - ▶ スポット検索
 - ...



特に、経路探索エンジンに必要なデータ量が数十～数百GB
Auto Scalingの足かせに・・・

NAVITIMEのAWS移行

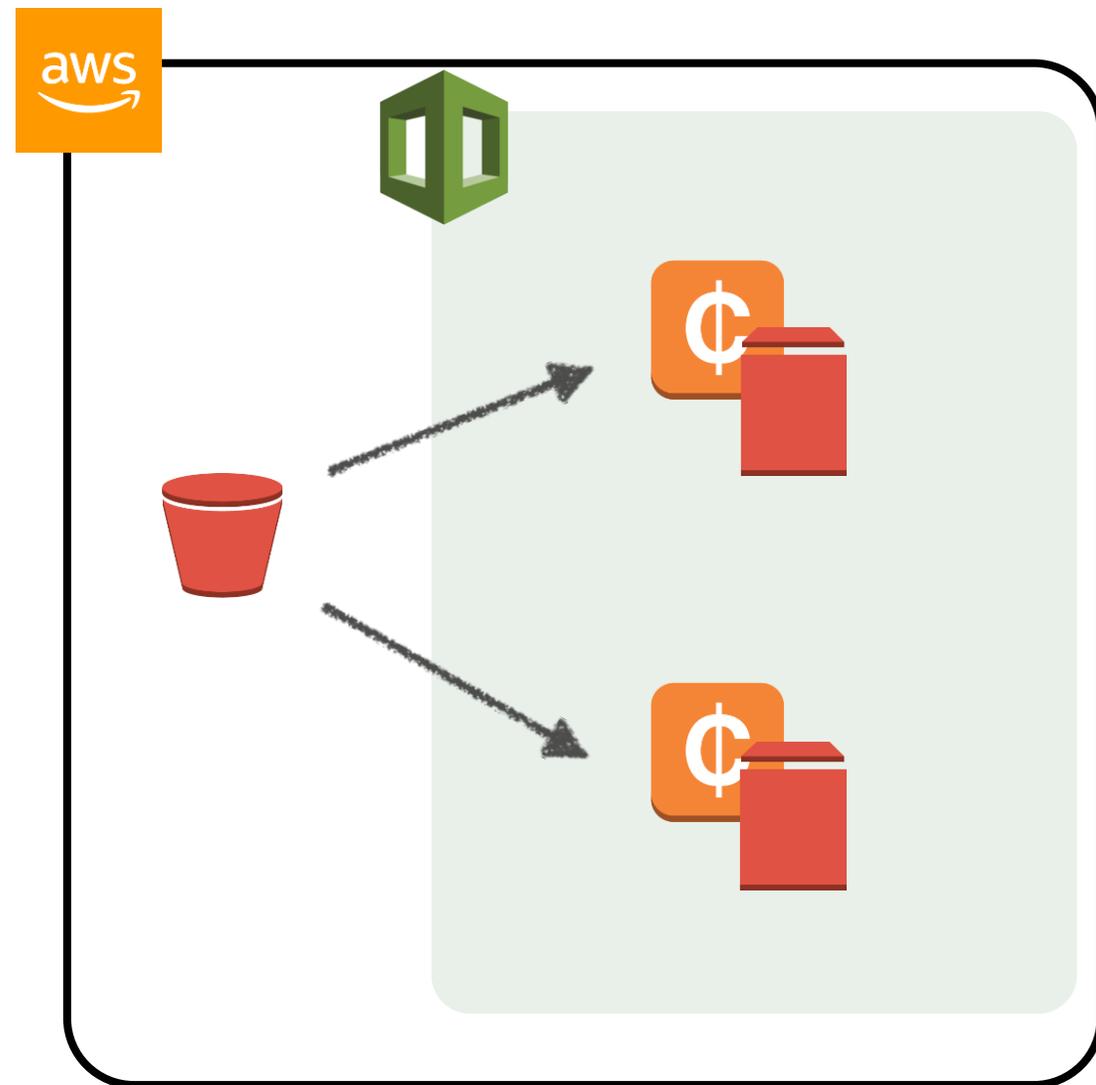
- ▶ 大容量データのデプロイ
 - ▶ 80～100GB程度のデータを数日に1回デプロイ
- ▶ デプロイ方法
 - ▶ S3からダウンロード
 - ▶ EBSスナップショットからの復元
 - ▶ EFSなどのファイル共有
 - ▶ その他

NAVITIMEのAWS移行

- ▶ 大容量データのデプロイ
 - ▶ S3からダウンロード
 - ▶ EBSスナップショットからの復元
 - ▶ どちらも初回アクセス時に遅くなる (EBSの仕様)
 - ▶ EFS
 - ▶ 東京リージョンまだ・・・
 - ▶ その他
 - ▶ EBSをプールさせておきスケールアウト時にアタッチする

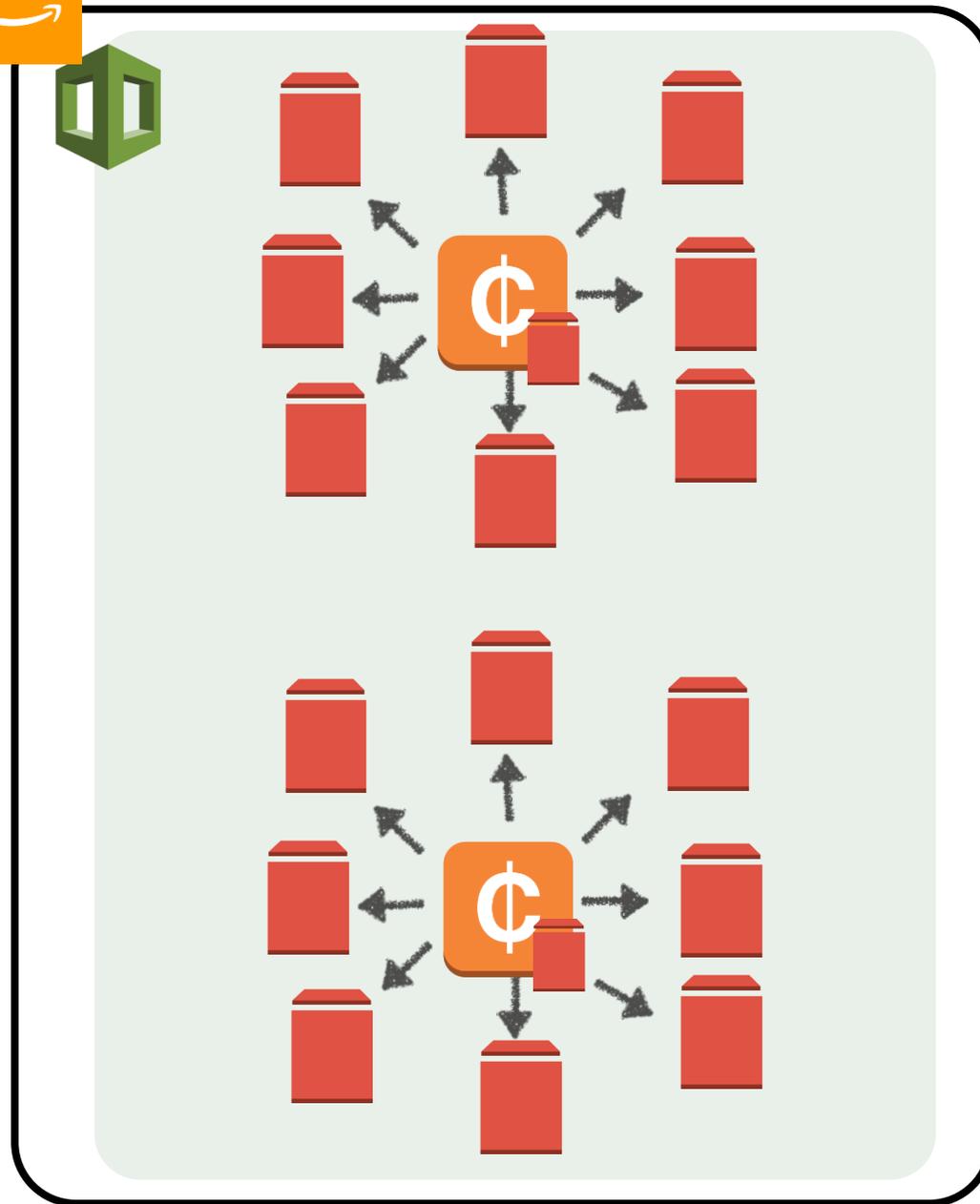
NAVITIMEのAWS移行

- ▶ 大容量データのデプロイ
 - ▶ 予めリリースデータをMasterEBSに転送する。
 - ▶ 初回だけ全件転送、2回目以降は差分転送



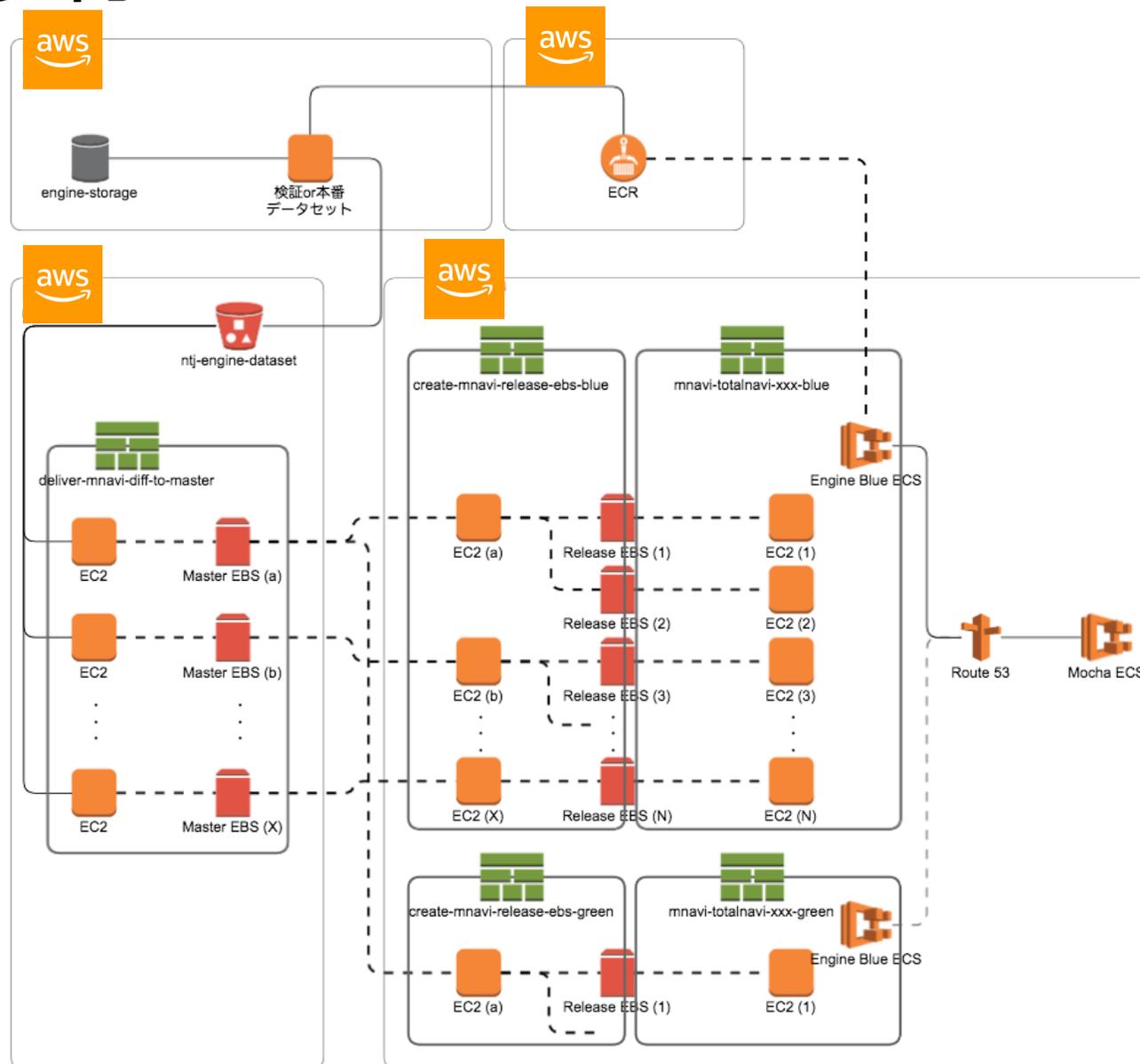
NAVITIMEのAWS移行

- ▶ 大容量データのデプロイ
 - ▶ インスタンスに作成したEBSと複数のリリース用EBSボリュームをアタッチし、並列でcopyを実行
 - ▶ デプロイされたEBSをプールしておき、AutoScaling時にインスタンスにアタッチ（EC2のUserData）



NAVITIMEのAWS移行

▶ 全体構成

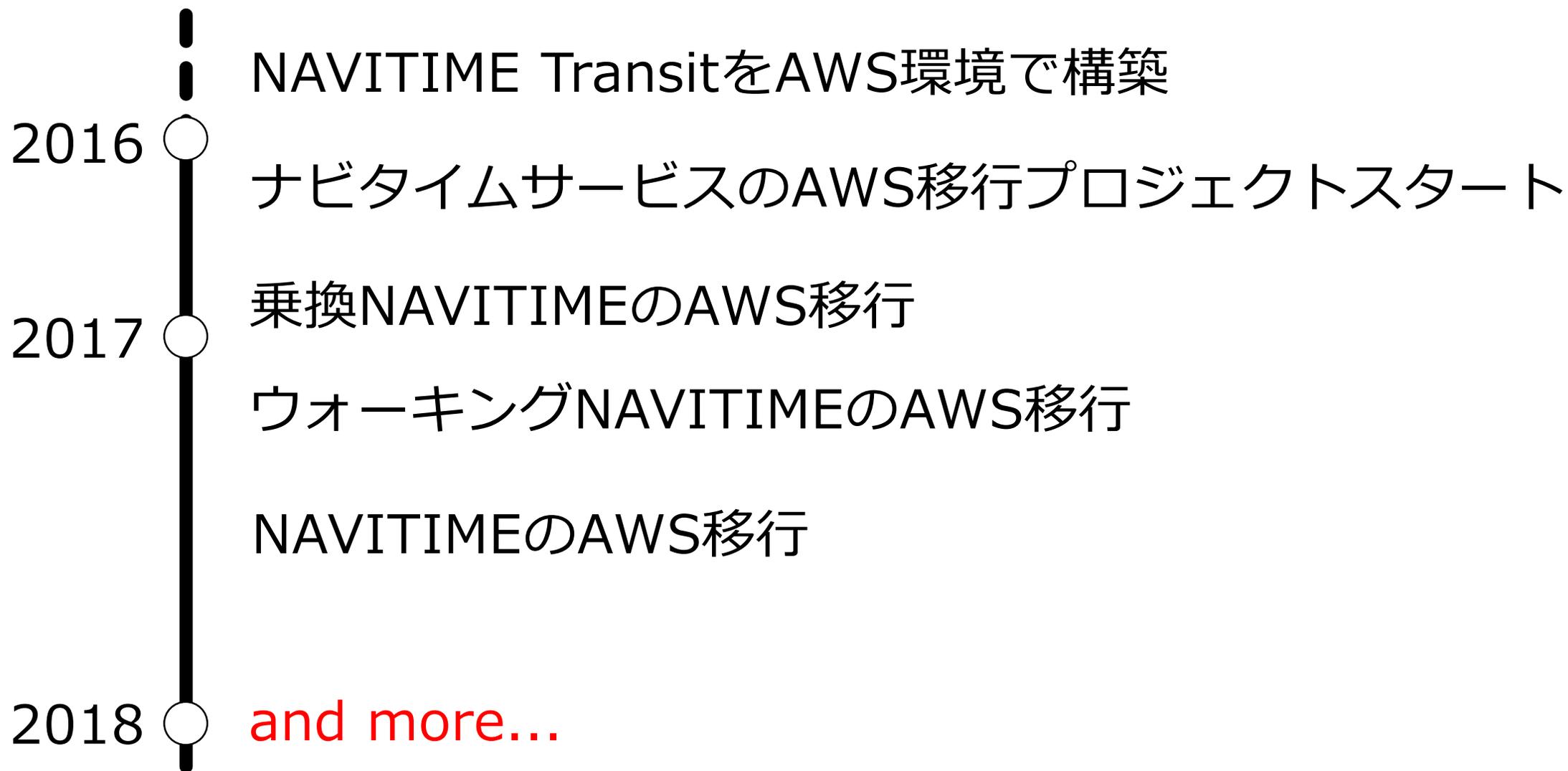


NAVITIMEのAWS移行

- ▶ 移行後3ヶ月経過
 - ▶ 安定稼働
 - ▶ 大きな障害なく稼働
 - ▶ 運用コストの削減
 - ▶ アクセスが増加する大型連休前のサーバ増設不要
 - ▶ オートスケールで対応



ナビタイムサービスのAWS活用



ナビタイムサービスのAWS活用

公共交通



NAVITIME



乗換
NAVITIME



バス
NAVITIME



こみれぽ

ドライブ



ドライブ
サポーター



カーナビタイム



トラックカーナビ

ツーリング



ツーリング
サポーター



自転車
NAVITIME

海外



NAVITIME
for Japan
Travel



NAVITIME
Transit

トラベル



NAVITIME
Travel



Plat by
NAVITIME

ヘルスケア



ウォーキング
NAVITIME

PC/SPブラウザ



ナビタイムサービスのAWS活用

- ▶ 既存サービスのAWS移行を計画
- ▶ Fargateなどを用いて更なる効率化
- ▶ AWSを用いた経路探索エンジンの高速化
 - ▶ GPUの積極活用



AWS Fargate



ナビタイムサービスのAWS活用

▶ AWS移行の効果

- ▶ サーバの輻輳が大幅減
- ▶ インフラの構成をコード化
- ▶ コンテナ化でアプリケーション管理の効率化



ナビタイムサービスのAWS活用

▶ AWS活用の副次効果

▶ コストの可視化

▶ オンプレ運用時には分かりにくかったインフラコストをサービスごとに明確化

▶ 明確化したことにより、**コスト意識の増大**

▶ インフラエンジニアの採用

▶ AWS採用を掲げクラウド化を前面に打ち出したところ、**インフラ部門に関心を持つエンジニアや配属希望者が急増し、人材確保の悩みが解消**

GPUでの超並列経路探索の取組み

アジェンダ

- ▶ 結論
- ▶ 自己紹介
- ▶ GPUと経路探索の相性
- ▶ GPU版 経路探索プロトタイプ
- ▶ 処理時間比較 GPU vs CPU
- ▶ P3インスタンスの威力
- ▶ まとめ

結論

結論

- ▶ GPUでは不向きと言われている処理でもGPUを活用できた
- ▶ 近年のGPUの進化により、かなり汎用的な処理でも手軽に高速化できる
- ▶ AWS GPUインスタンスは低コストで手軽に試しやすく、インスタンスタイプを変えるだけで最新GPUを利用できる

GPUインスタンス
"機械学習以外でも"
どんどん使いましょう！！

自己紹介

自己紹介

吉濱 誠

株式会社ナビタイムジャパン

開発部

ACTS(研究開発) シニアエンジニア

2016

サーバサイドAPI開発、経路探索エンジンの開発

2017

Pepperアプリ開発や、Cognitive Serviceを積極的に活用したチャットボットの開発を担当

GPGPUを利用した超並列探索アルゴリズムのプロトタイプ開発を担当



経路探索にGPUを利用する

- ▶ 月間ユーザ数の増加
 - ▶ 3700万人(2017年6月末) → 4100万人(2017年12月末)
 - ▶ 処理時間の高速化が必要
- ▶ 経路品質の向上
 - ▶ 既存アルゴリズムでは出せない経路が出るようになる

品質、処理速度の向上のため、経路探索をパラダイムシフトする

GPUと経路探索処理の相性は…？

よく聞くGPU利用事例

よく聞くGPU利用事例

機械学習ライブラリから
GPUリソースへアクセス可能に！

よく聞くGPU利用事例

GPUを利用することで
機械学習の学習時間をN倍高速化！

つまり配列計算

配列計算の特徴

- ▶ 要素数が多い
- ▶ 処理の粒度が小さい
 - ▶ 各要素に対して四則演算を行う程度
- ▶ コアレスアクセスしやすい
 - ▶ 連続したメモリ領域にアクセスする

処理は単純だが数が膨大なもの
に向いている

GPUプログラミングでの一般論

- 条件分岐は避けましょう
- なるべく連続メモリにアクセスしましょう

経路探索処理は…？

経路探索処理の特徴

- ▶ 要素数が多い
 - ▶ リンク数は全国で2700万リンク
- ▶ 処理の粒度が大きい
 - ▶ 複数の要因から成るリンクのコスト計算
 - ▶ リンクを拡散するか、しないかの条件判断
 - ▶ 動的な交通情報、規制情報などの考慮
- ▶ ネットワークデータへランダムアクセス

経路探索処理の特徴

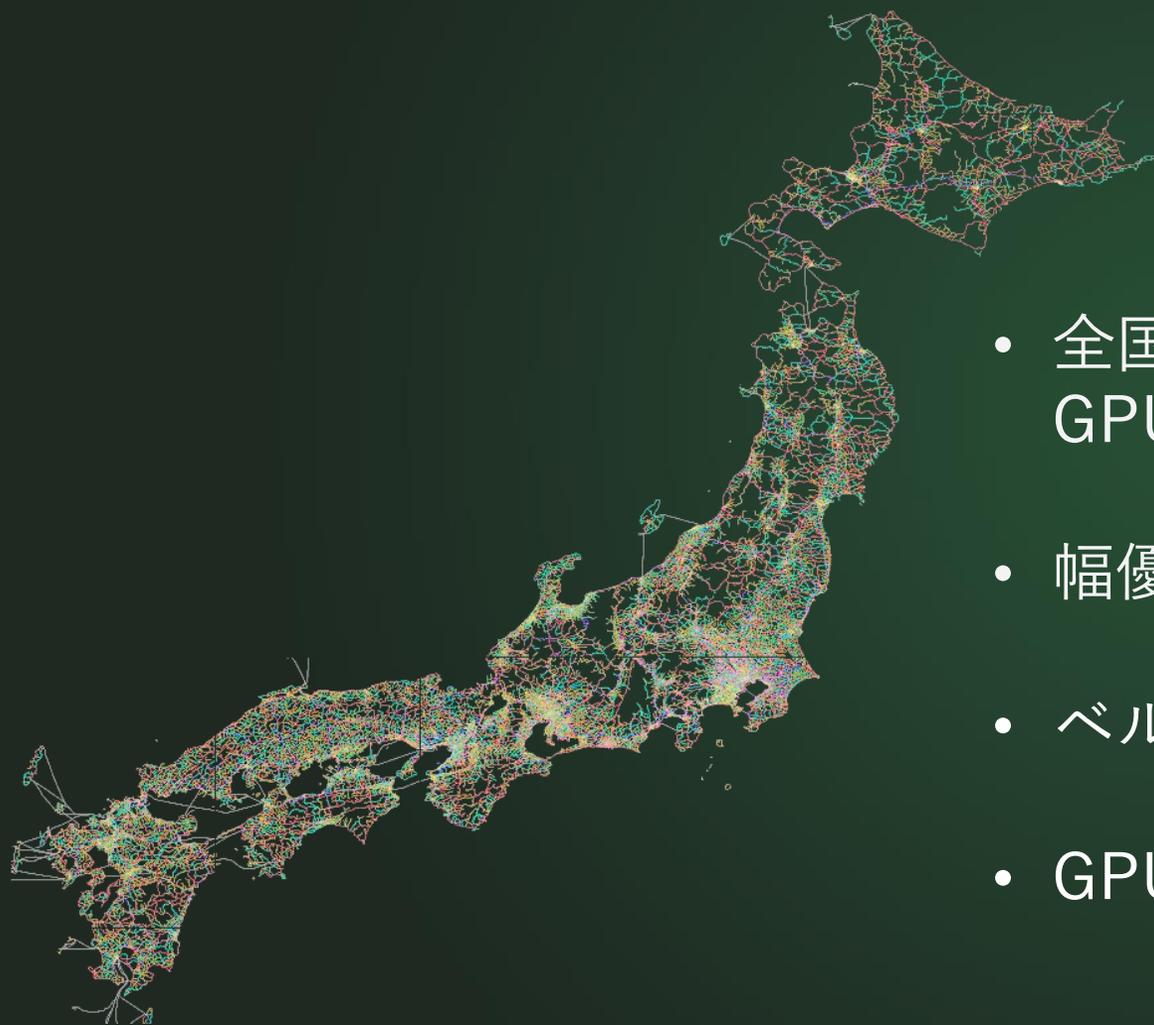
- ▶ 要素数が多い
 - ▶ リンク数は全国で2700万リンク
- ▶ 処理の粒度が大きい
 - ▶ 複数の要因による計算
 - ▶ リンクを単位とした条件判断
 - ▶ 動的な変数などの考慮
- ▶ ネットワークデータへランダムアクセス

アンチパターン
てんこ盛り

処理粒度の大きい処理を
GPUで高速化できるか

そもそも経路探索できるか

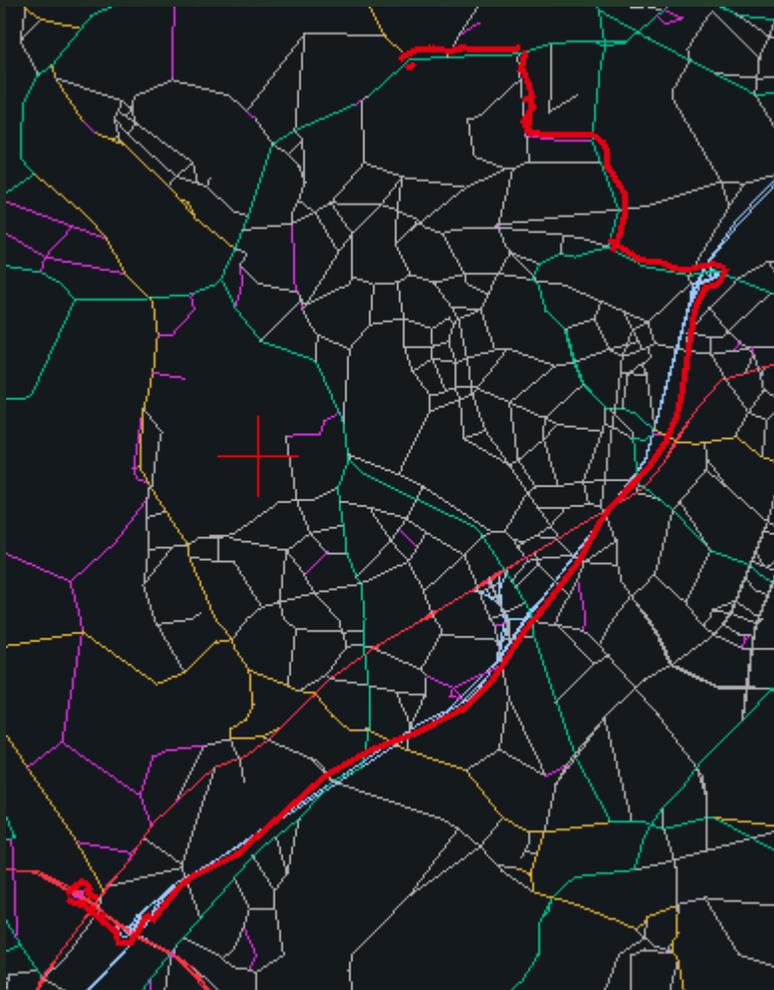
経路探索プロトタイプ概要



1リンク

- 全国のネットワーク(2700万リンク)を全てGPUに載せる(サイズは2.6GB)
- 幅優先探索
- ベルマンフォード法
- GPUインスタンス：[p2.xla \(Kepler\)](#)

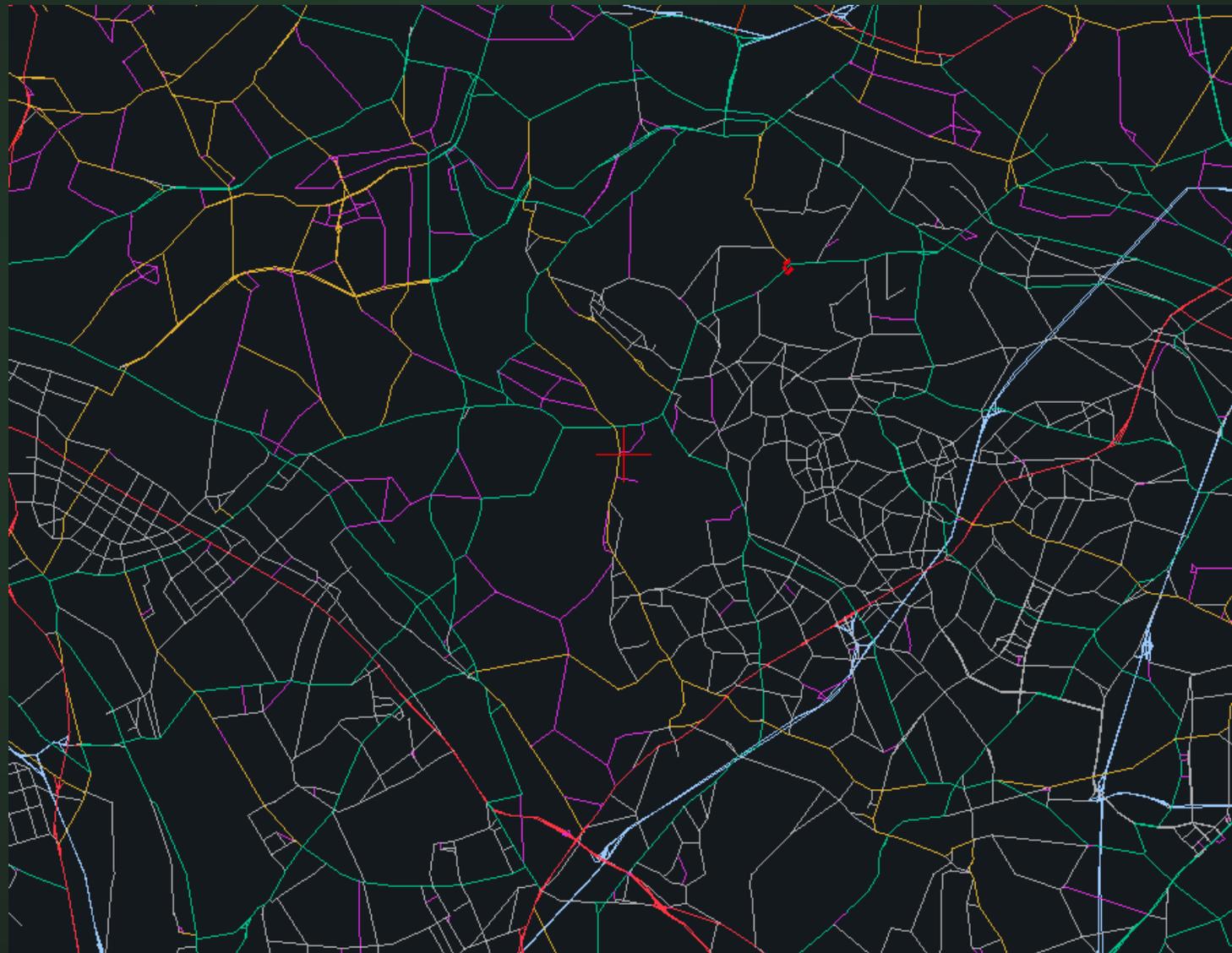
結果



探索できた

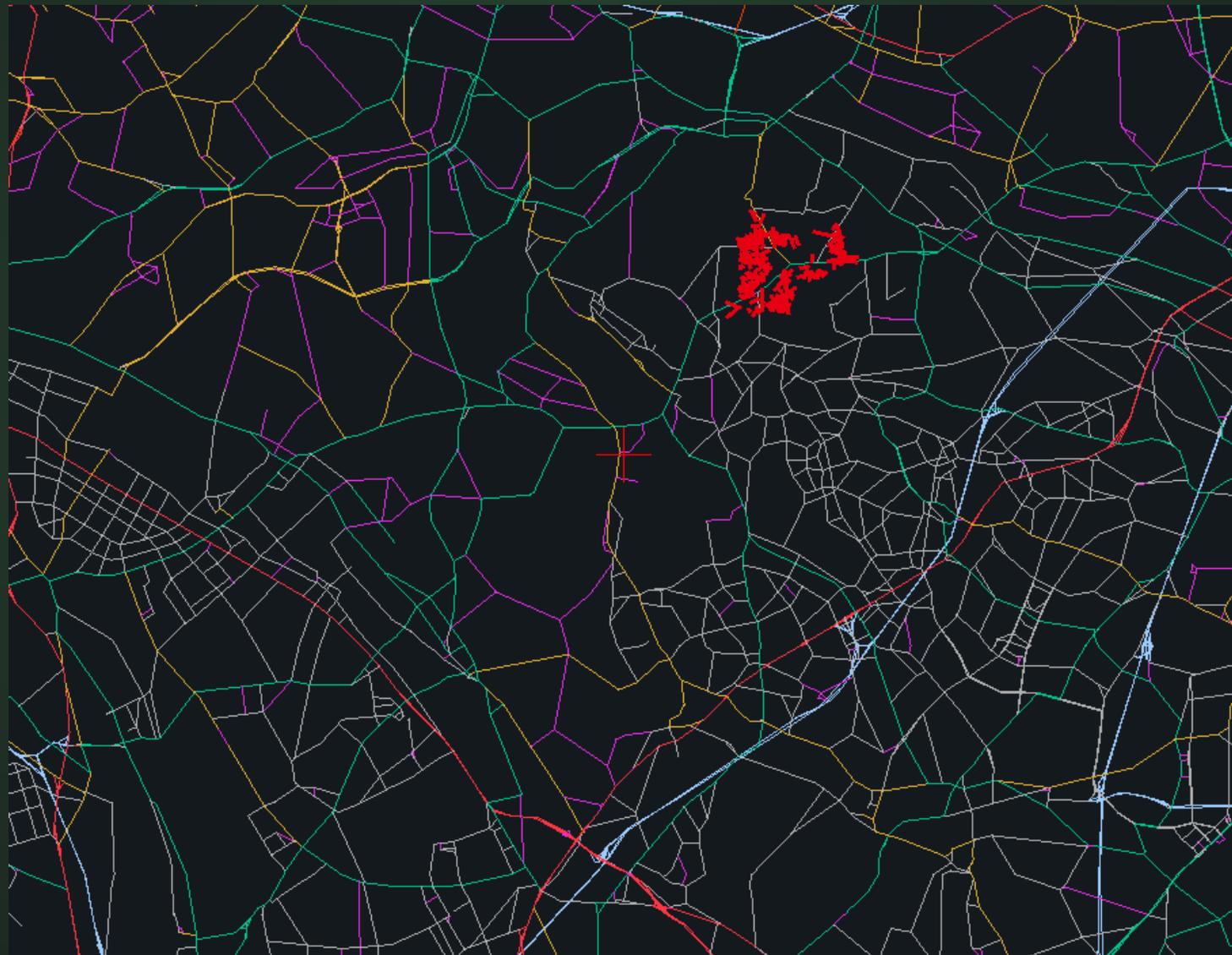
探索処理のようす

イテレーション：0



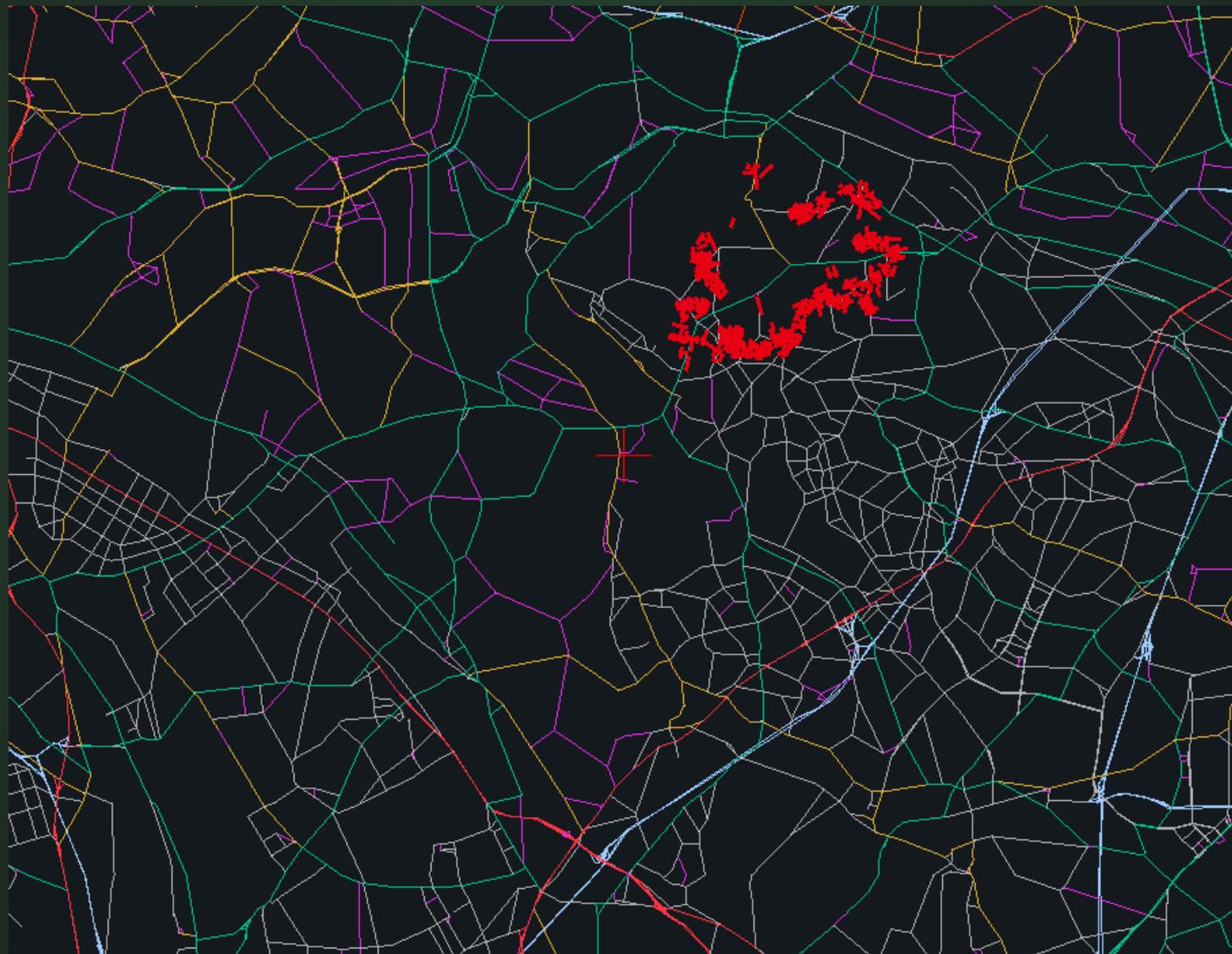
探索処理のようす

イテレーション：20



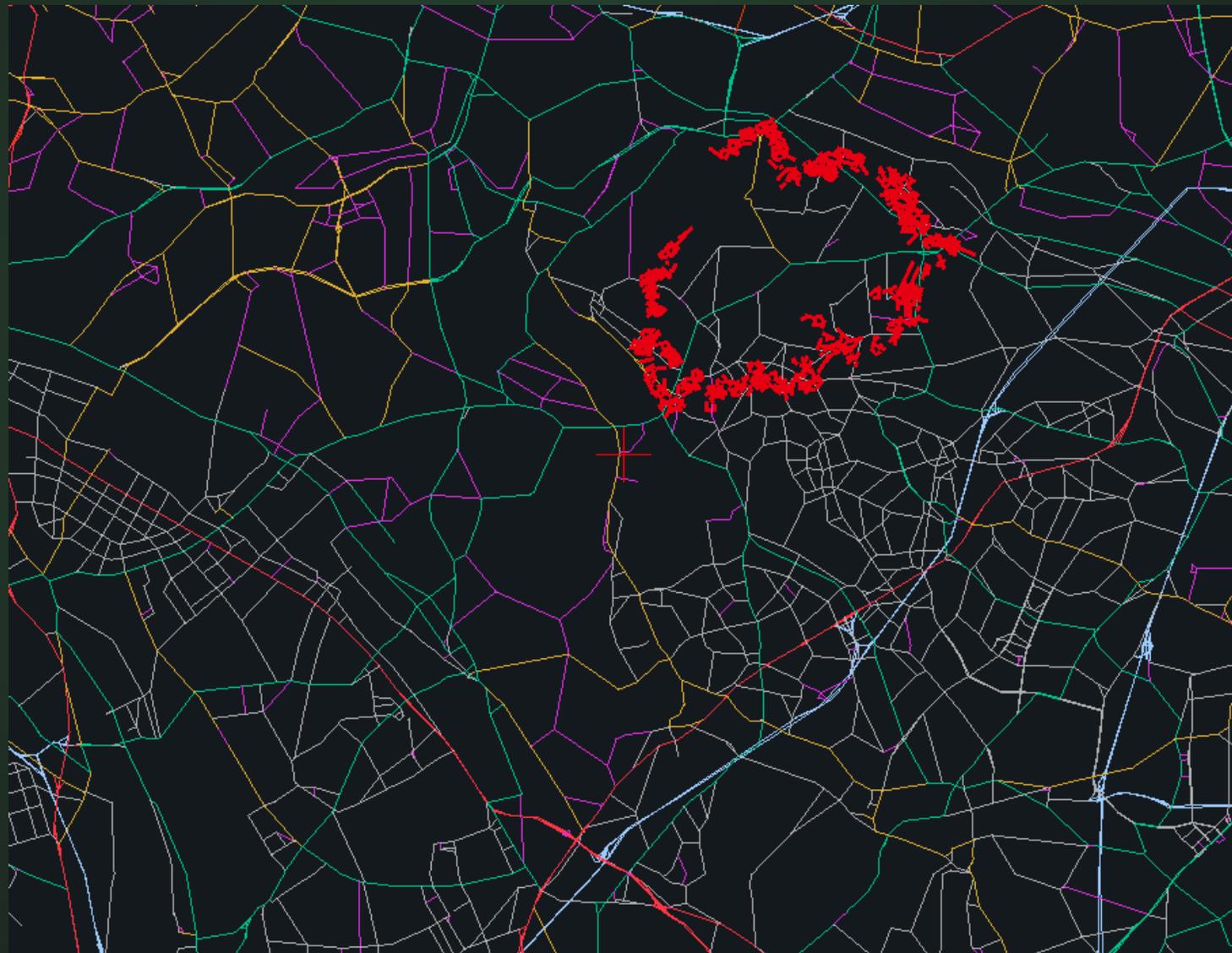
探索処理のようす

イテレーション：40



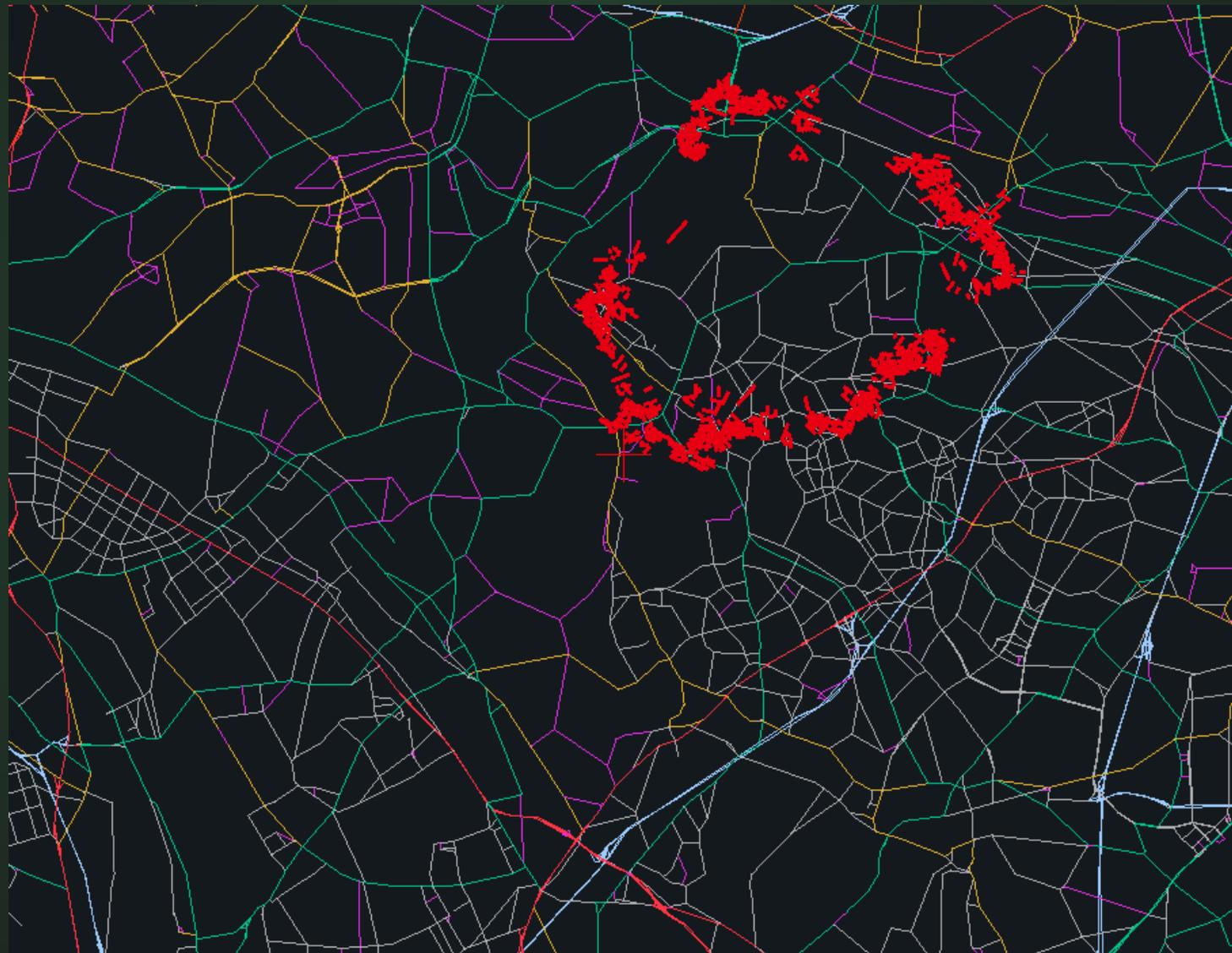
探索処理のようす

イテレーション：60



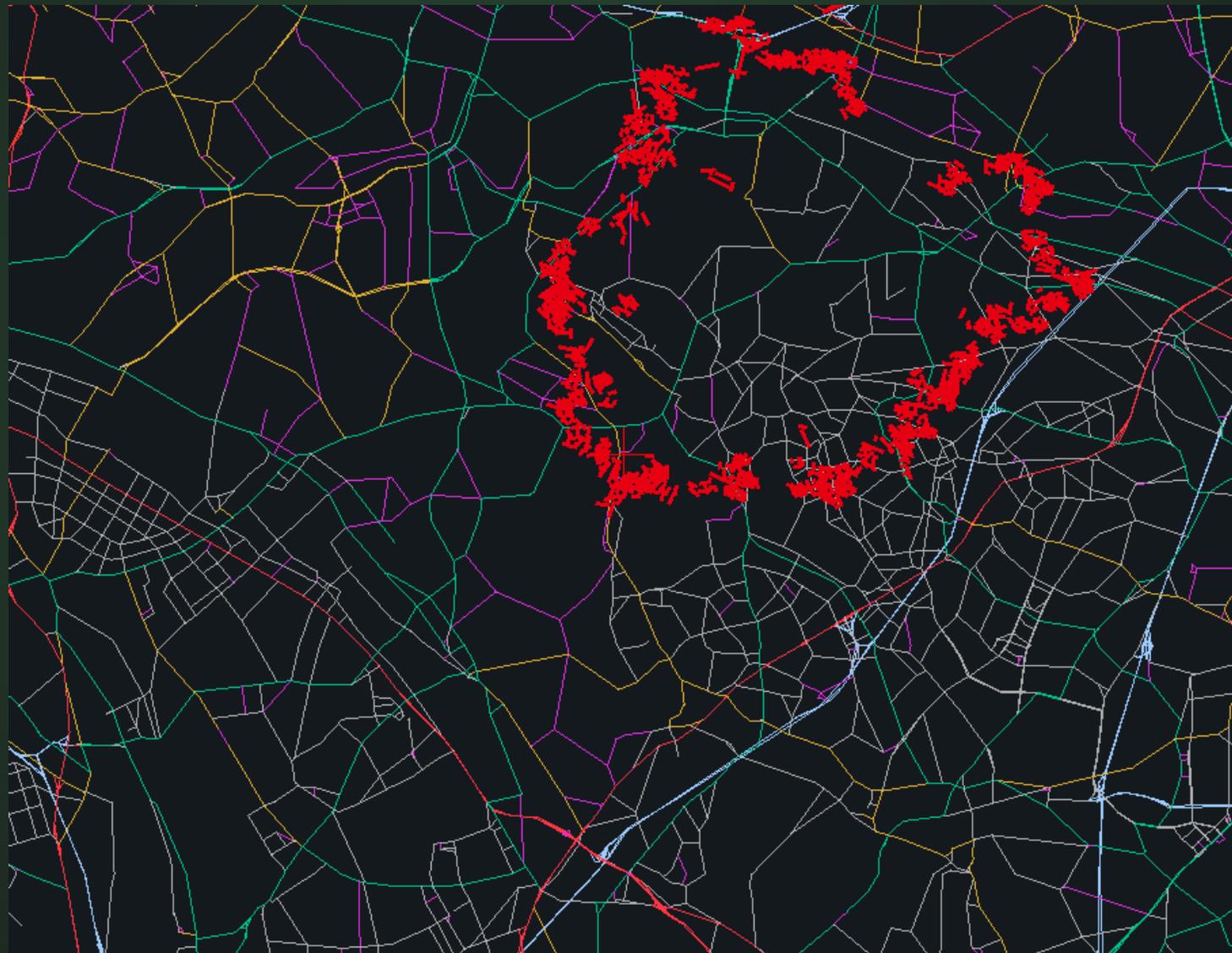
探索処理のようす

イテレーション：80



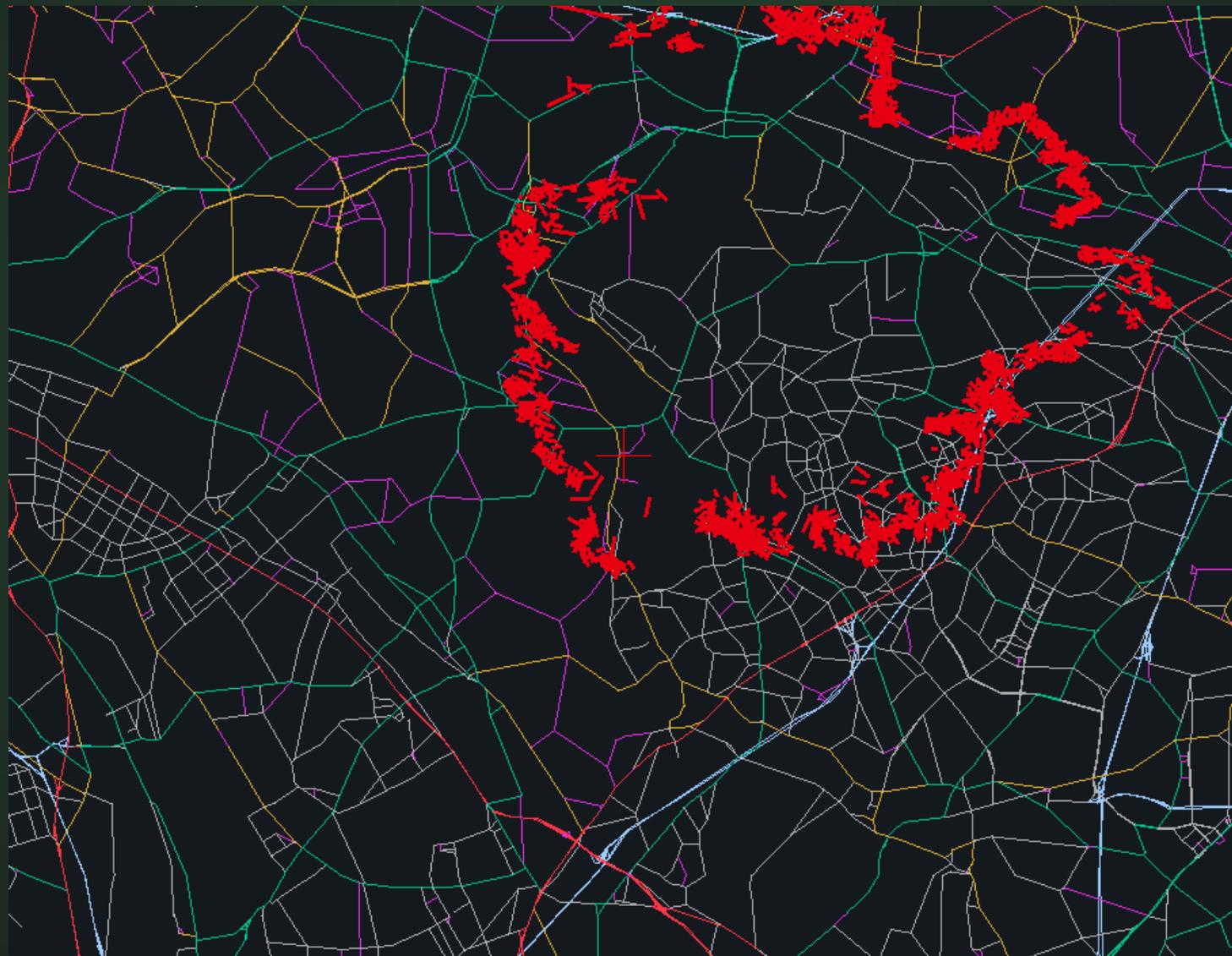
探索処理のようす

イテレーション：100



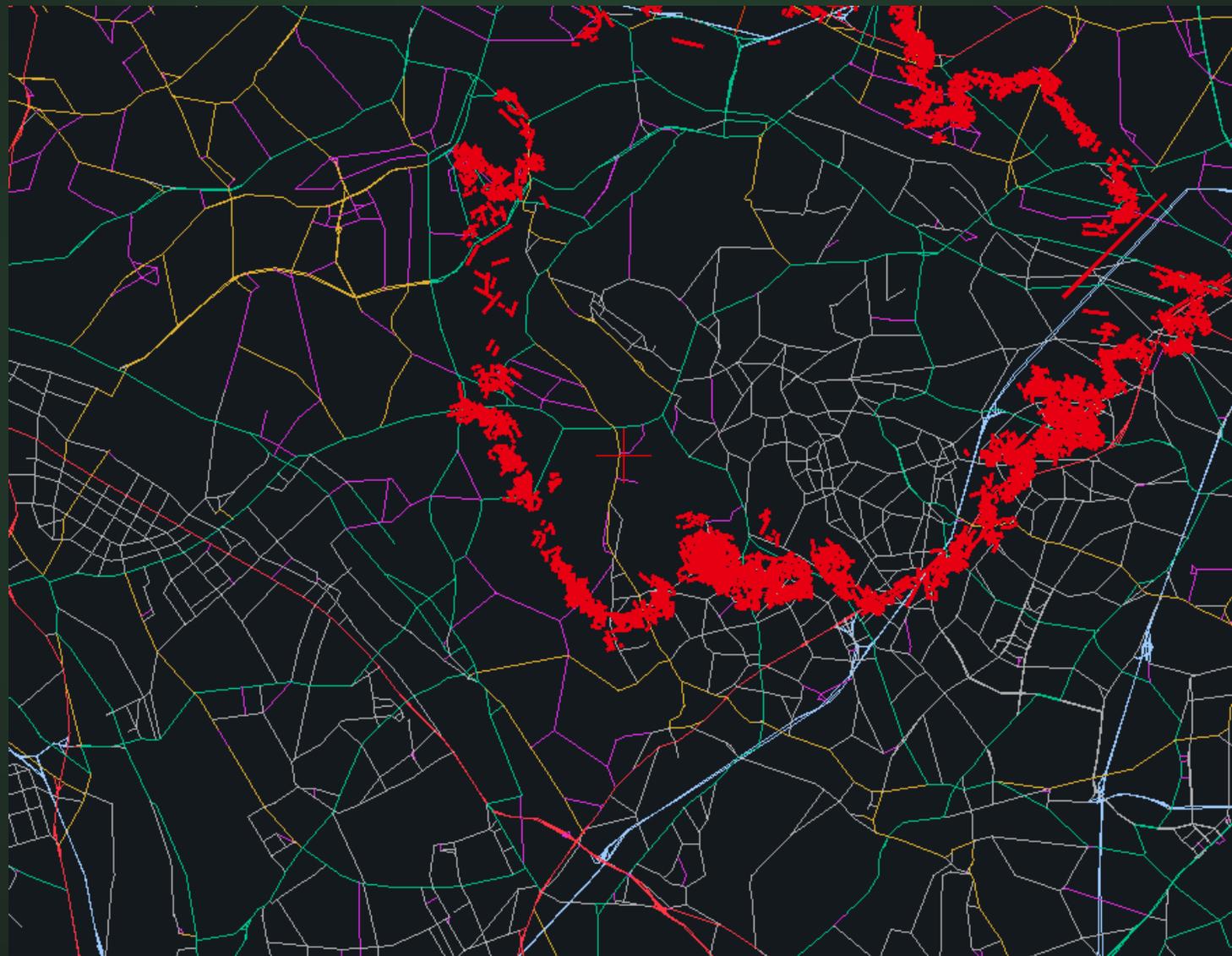
探索処理のようす

イテレーション：120



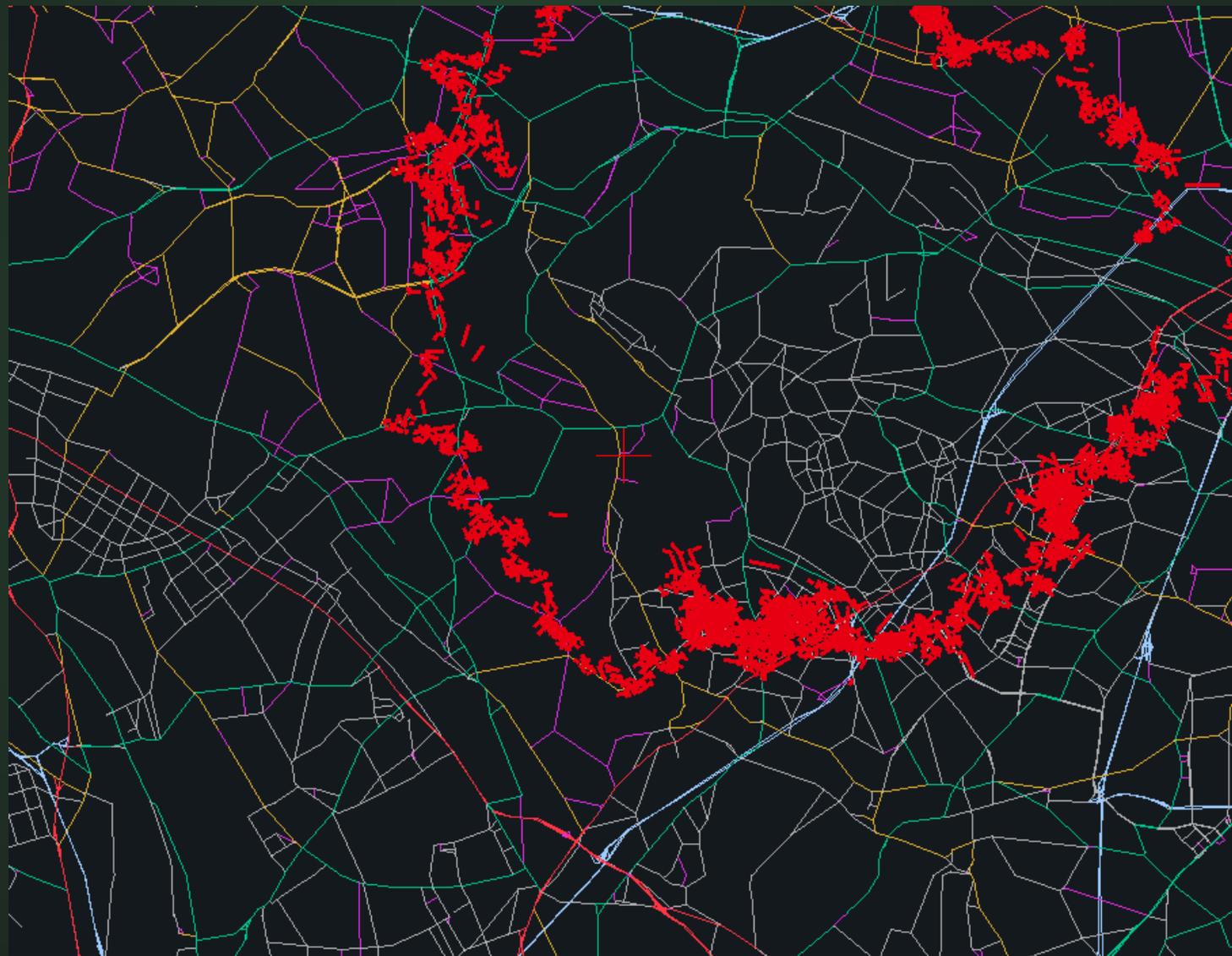
探索処理のようす

イテレーション：140



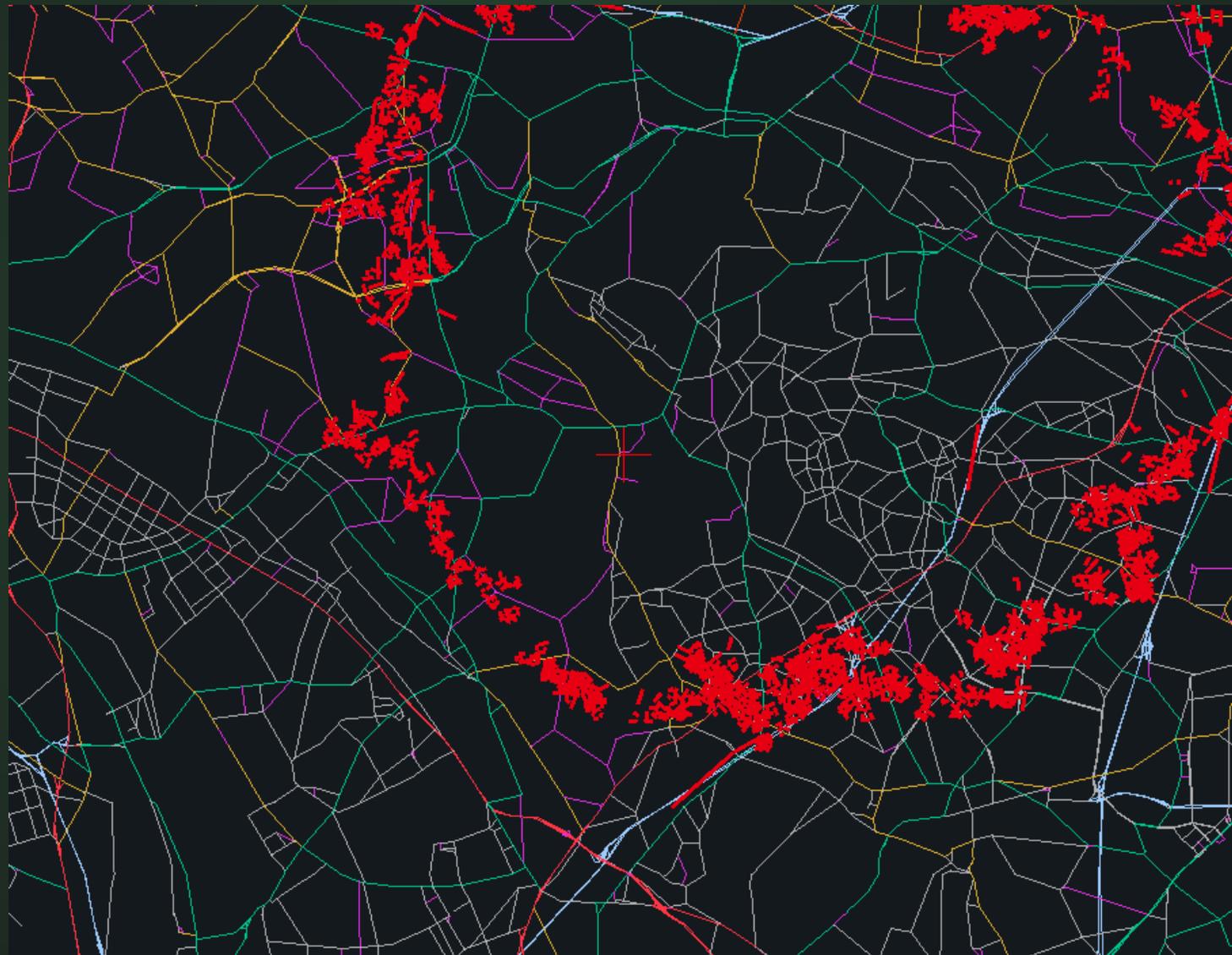
探索処理のようす

イテレーション：160



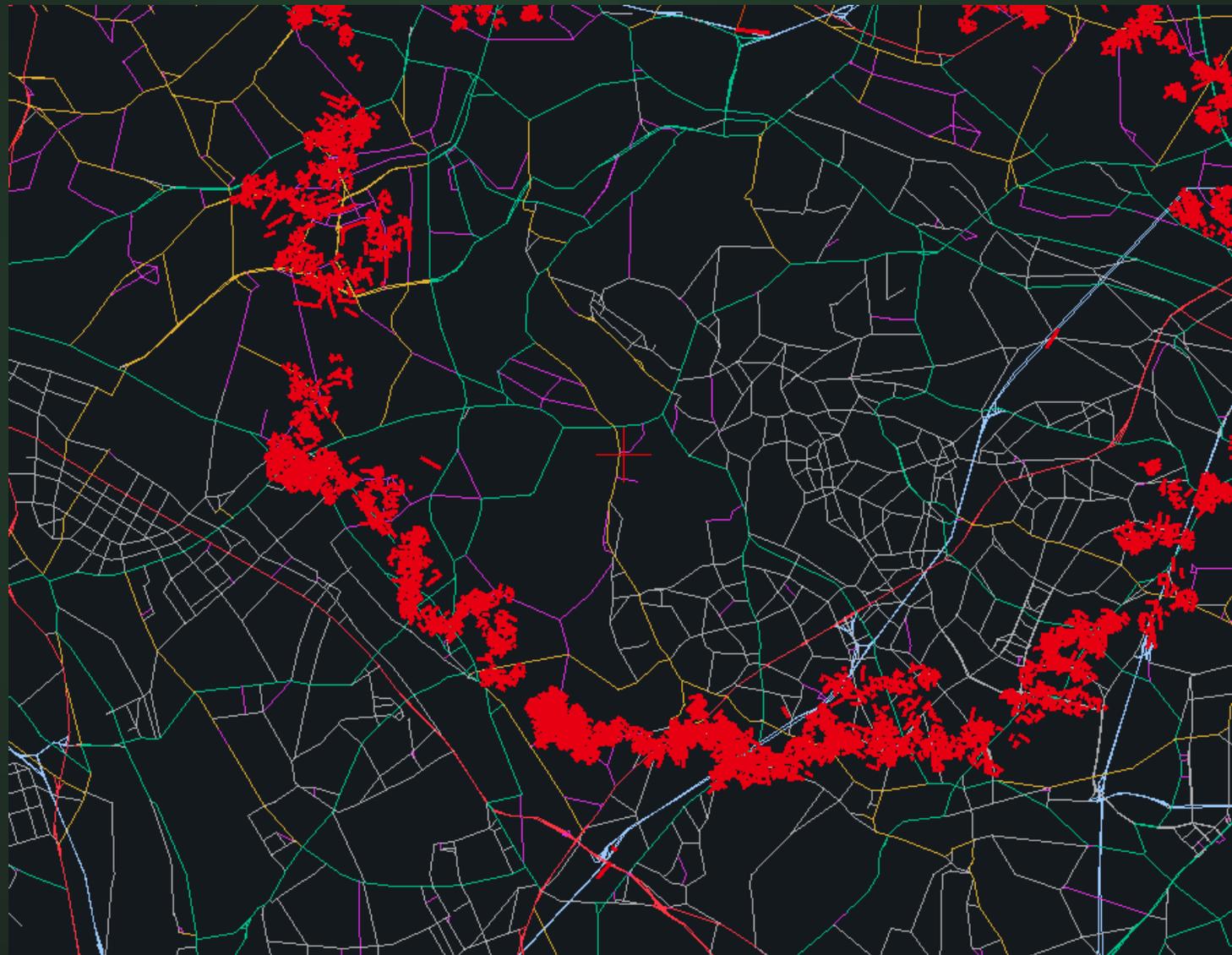
探索処理のようす

イテレーション：180



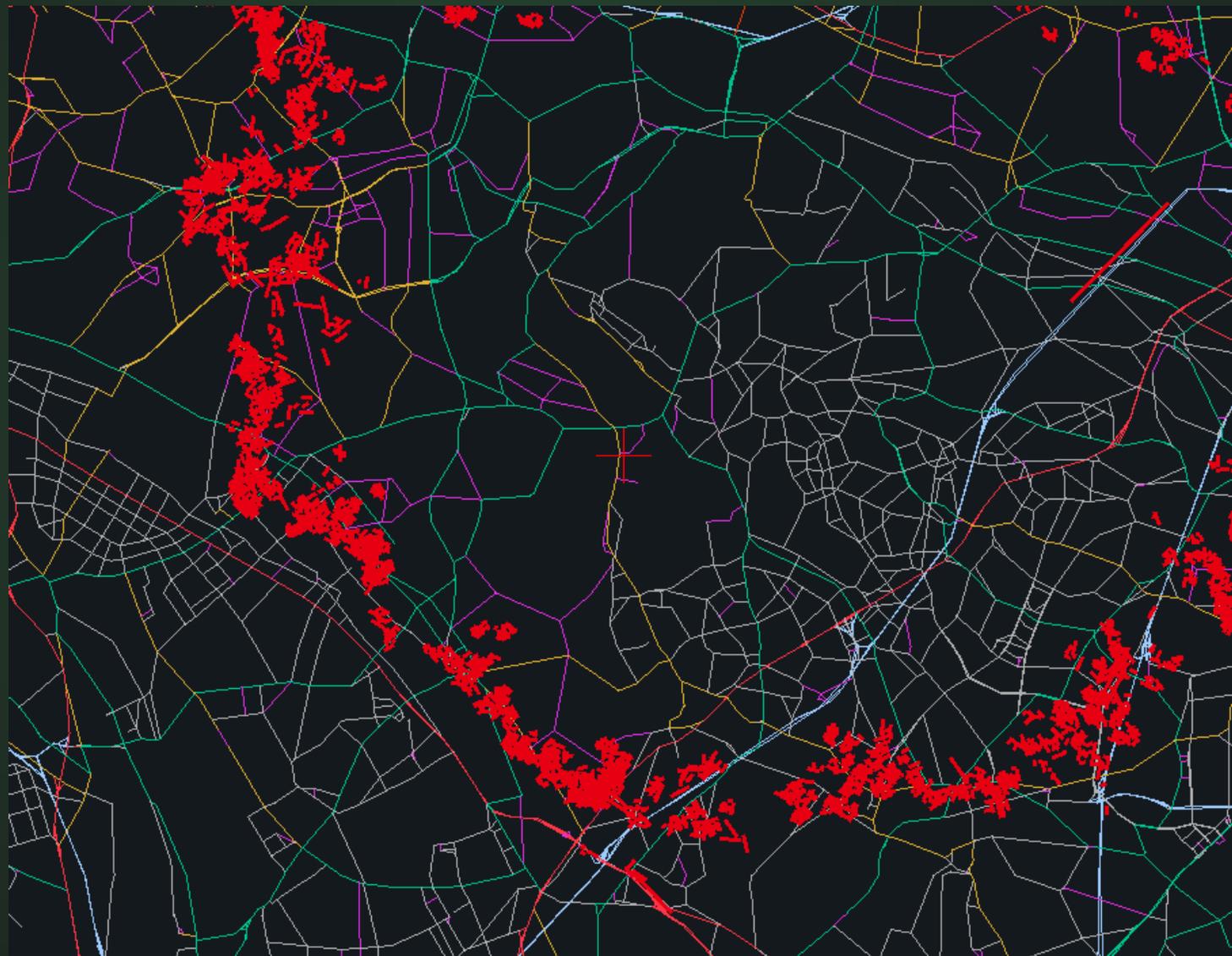
探索処理のようす

イテレーション：200



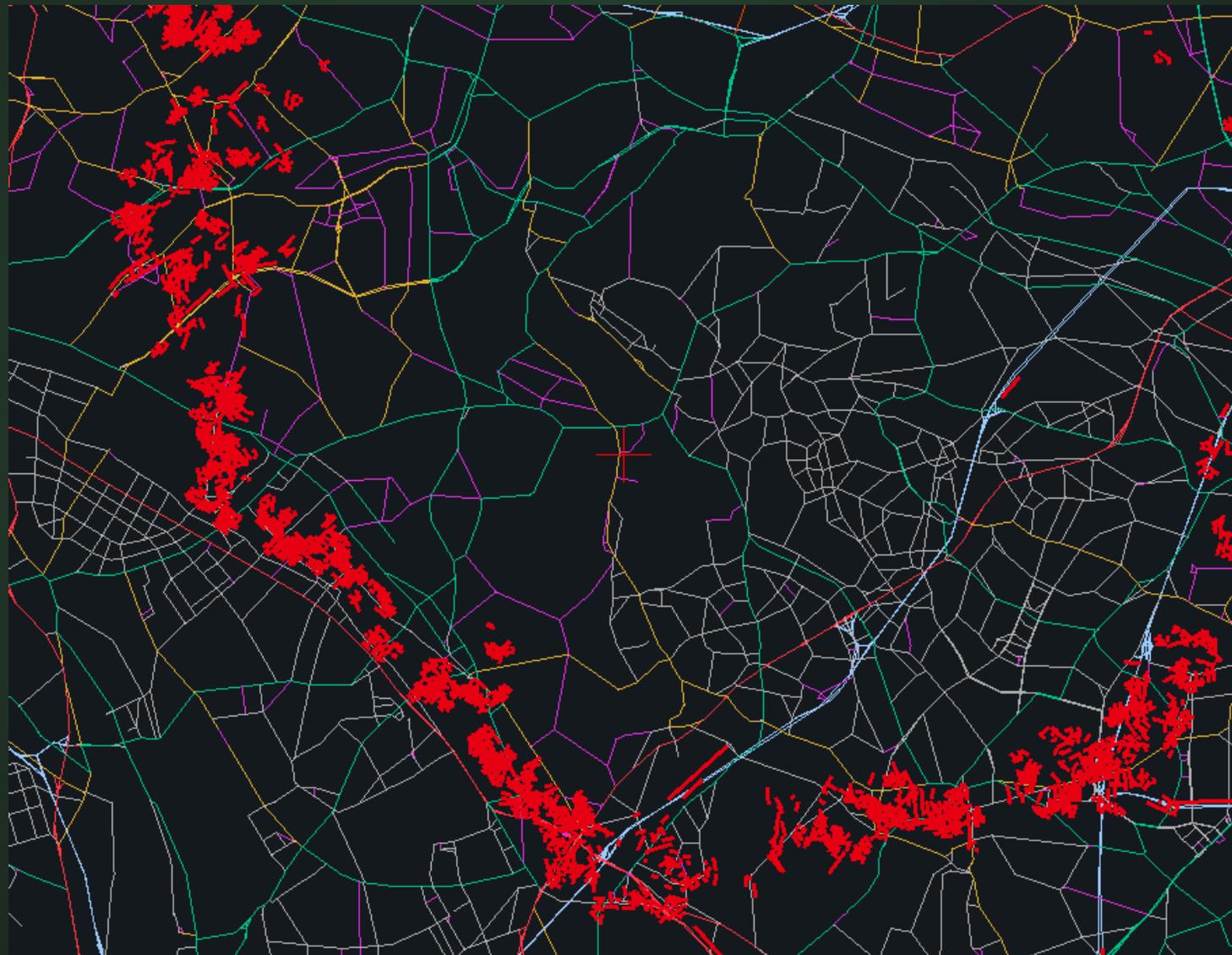
探索処理のようす

イテレーション：220

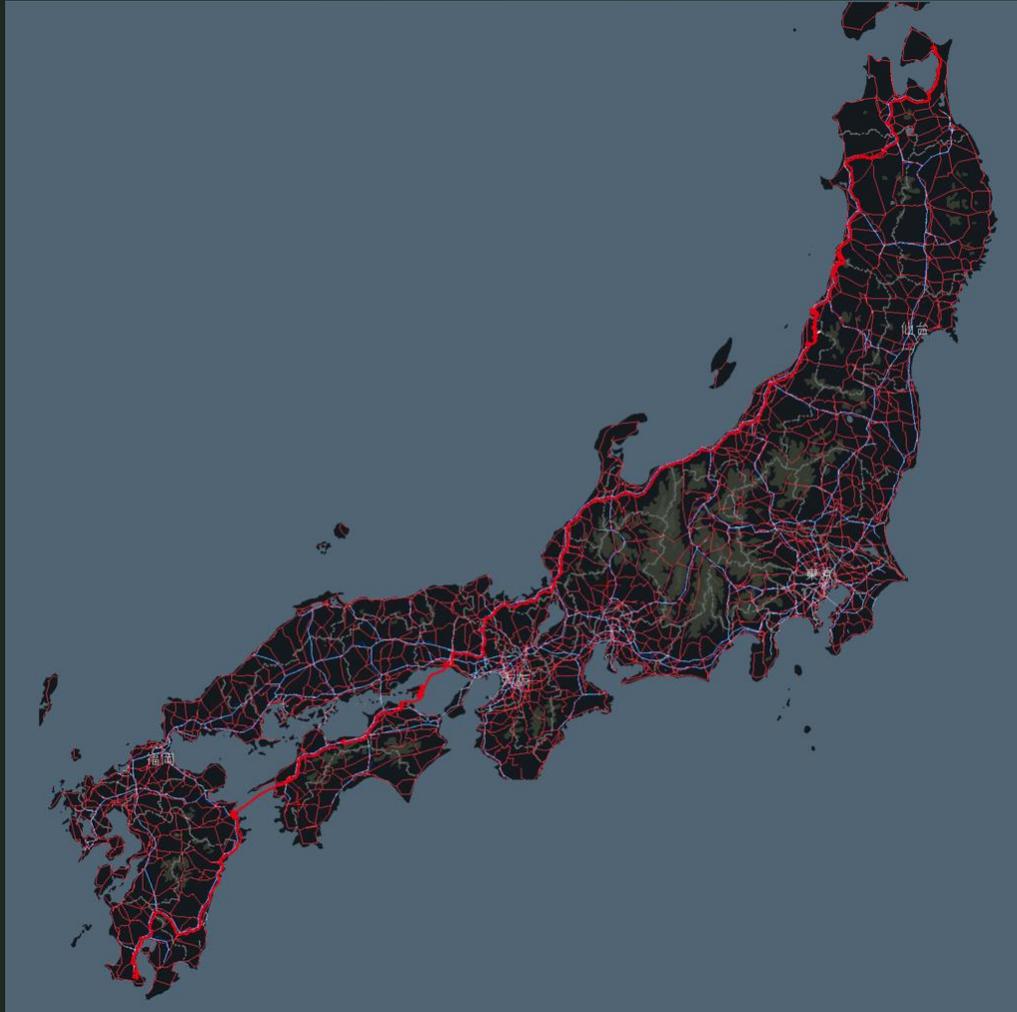


探索処理のようす

イテレーション：240



長距離の探索結果



長距離でも
探索できた

処理時間比較 GPU vs CPU

▶ CPU版は2通り

▶ ネットワークデータ全考慮版

- ▶ GPUと同等の条件

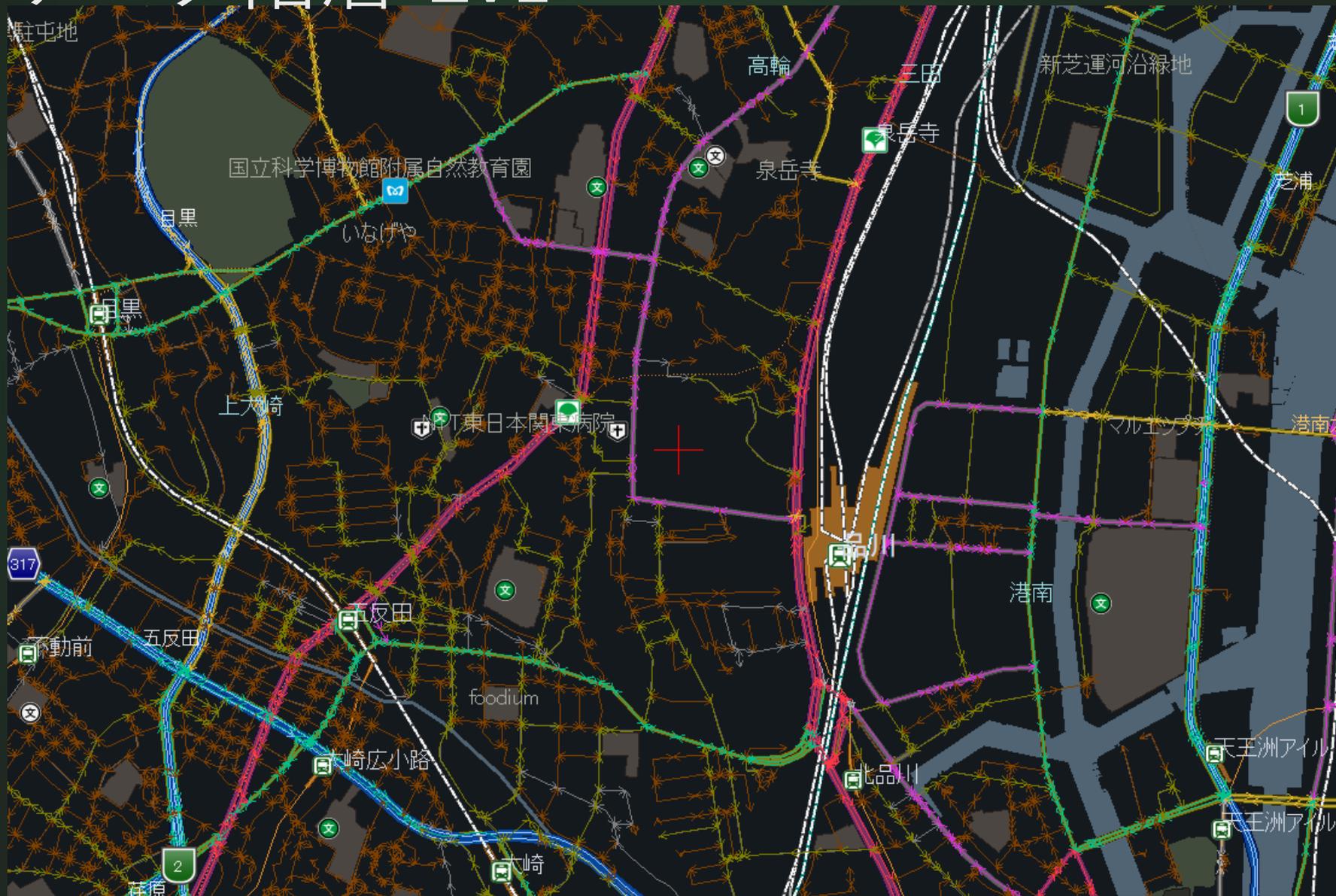
- ▶ 全てのネットワークデータを考慮するため、裏通りなどの経路まで出せる高品質版

▶ ネットワークデータ省略版

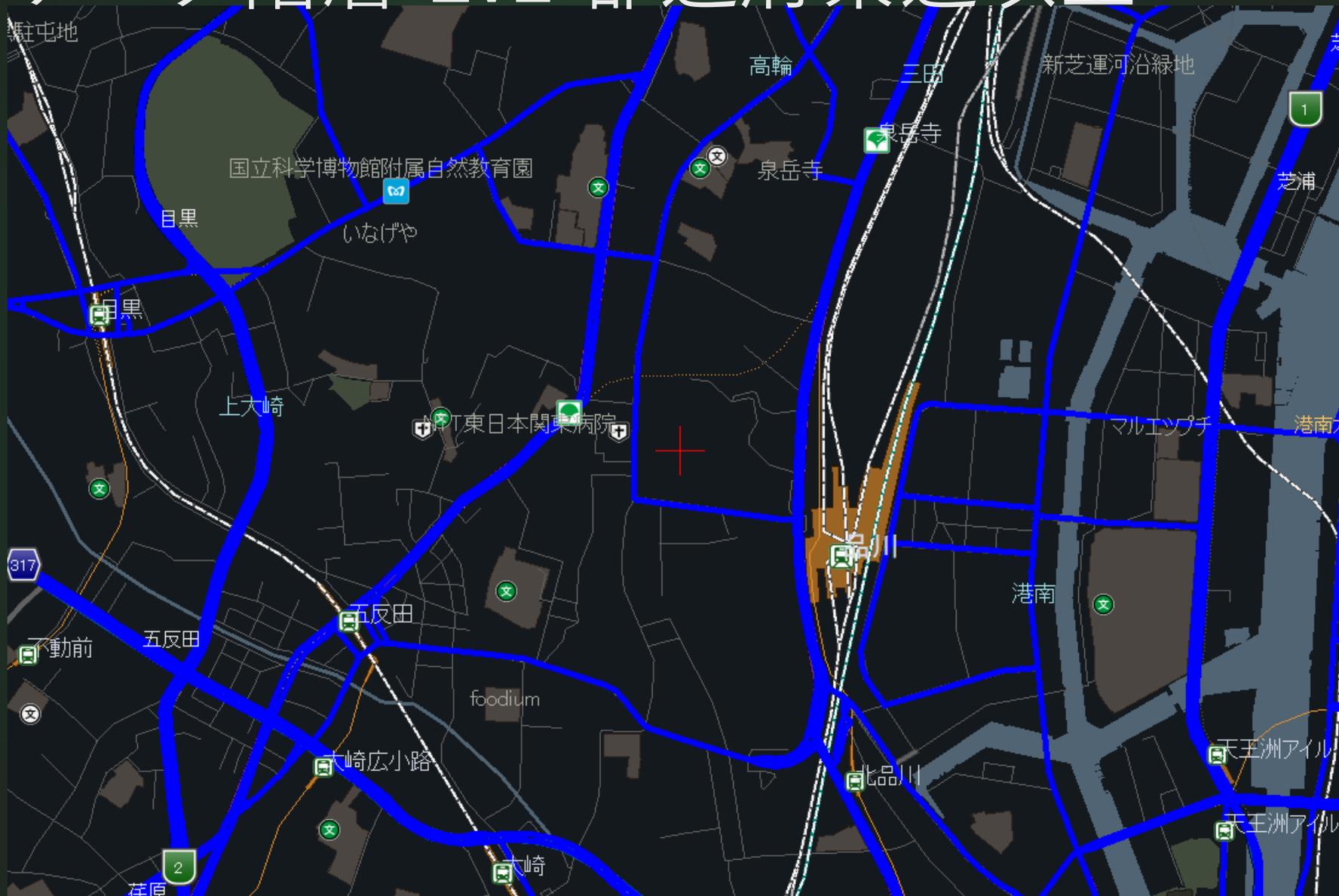
- ▶ 高速化のためネットワークデータを階層化し、探索距離に合わせて見るデータを減らしている

- ▶ 経路の途中で見えるネットワークデータが減るため、品質の向上に限界がある

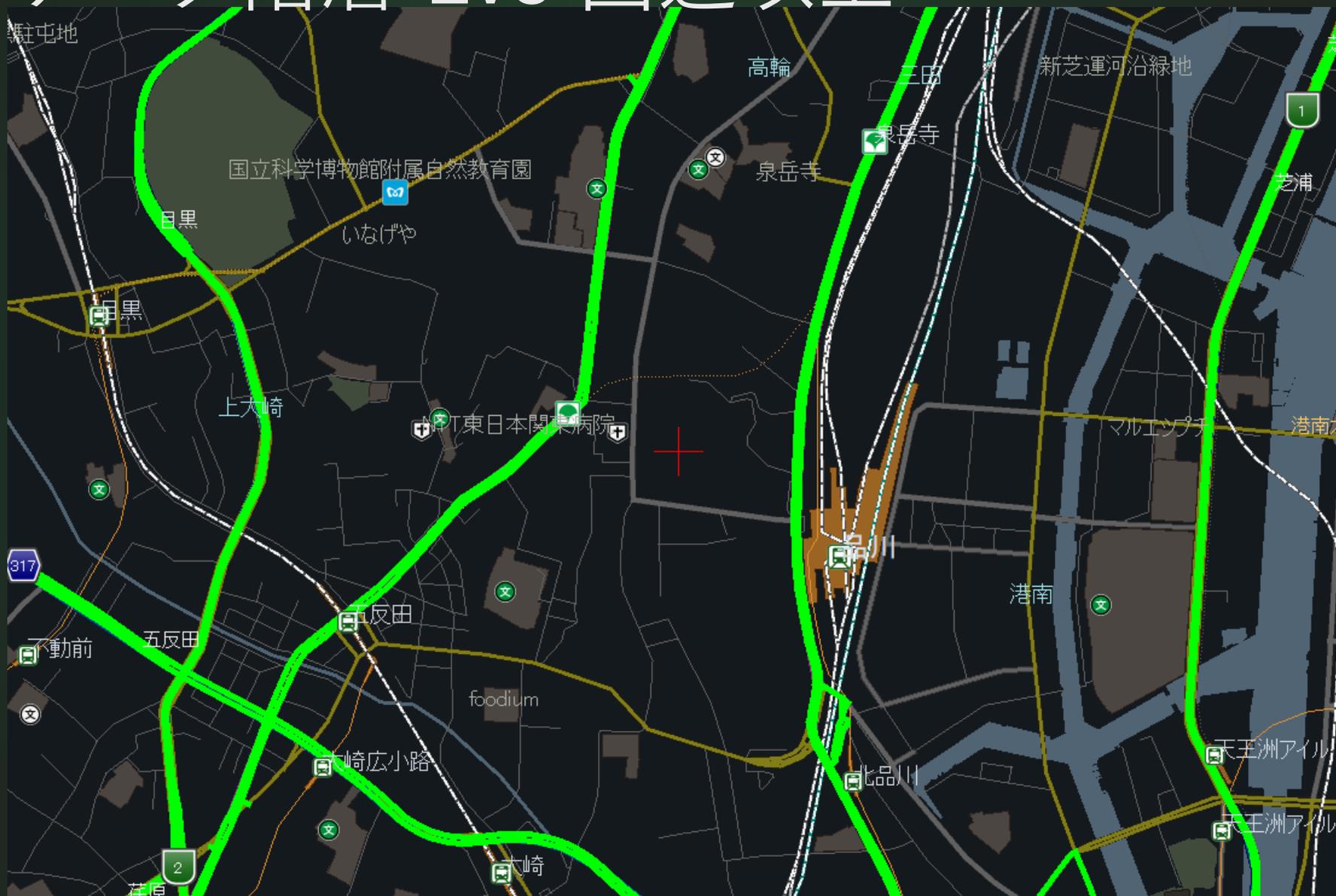
ネットワーク階層 Lv1



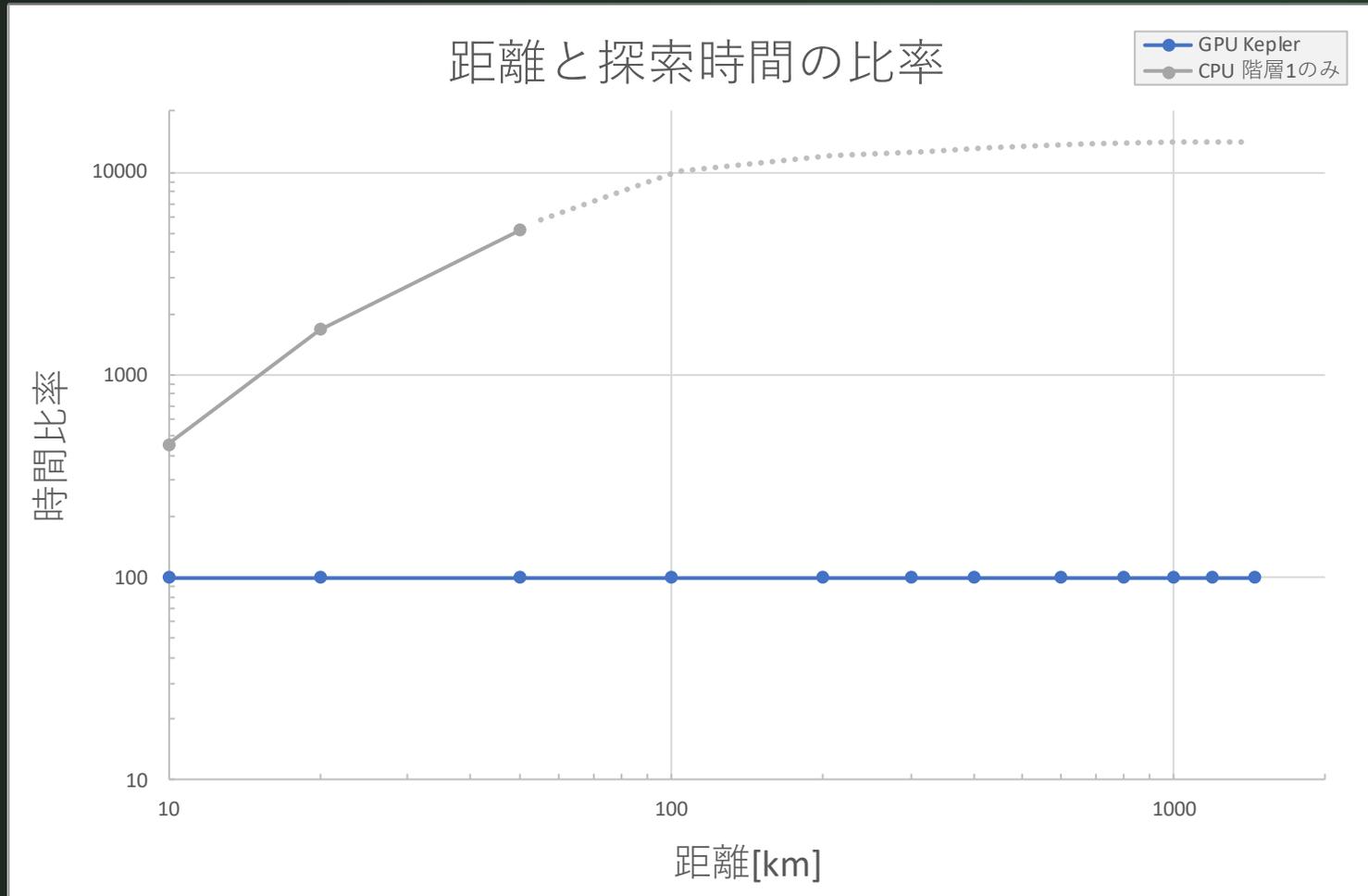
ネットワーク階層 Lv2 都道府県道以上



ネットワーク階層 Lv3 国道以上



GPU vs CPU (ネットワークデータ全考慮版)



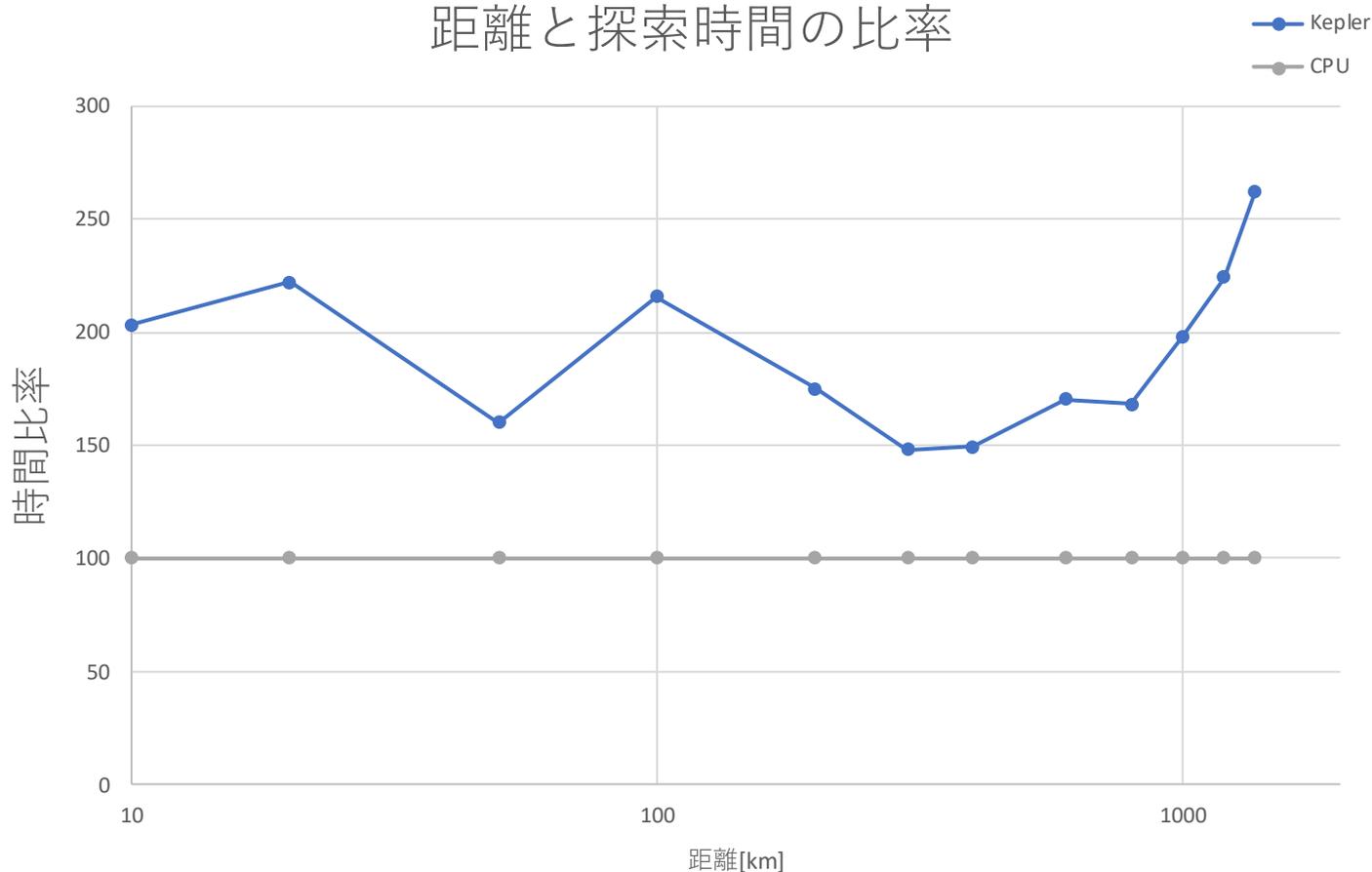
GPUと同じ条件
(ネットワークリンク
を全て考慮)

CPU版は距離50km
までしか計測できず

距離50kmの探索で
GPUが60倍速い

GPU vs CPU (ネットワークデータ省略版)

距離と探索時間の比率



さすがにネットワークデータ省略版にはそのままでは勝てなかった…

課題

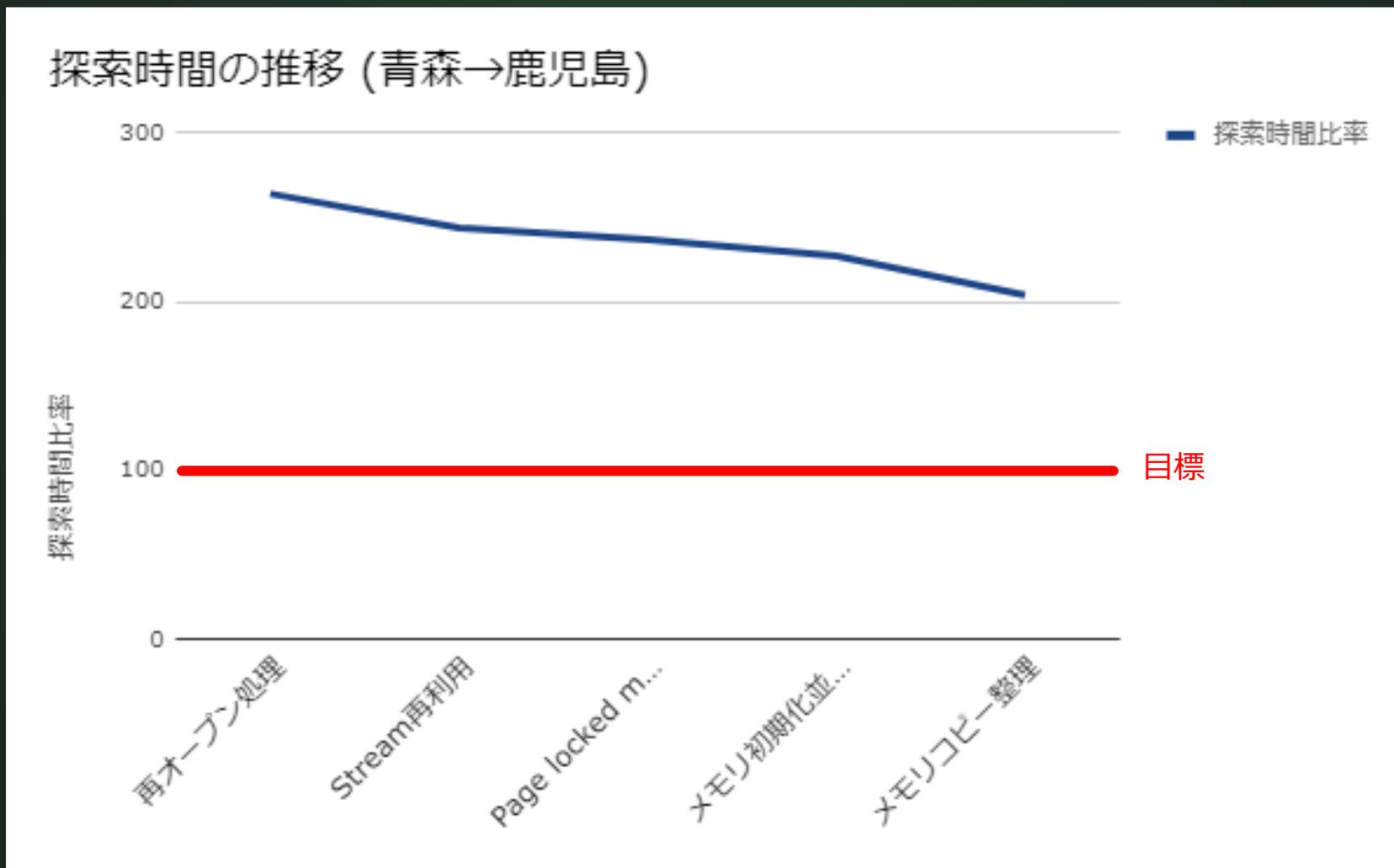
▶ 長距離での処理時間

▶ CPU版：100
GPU版：265

▶ 同じアルゴリズムをCPUで処理した場合と比較すれば10倍以上の高速化ができていますが、ネットワークデータを省略したCPU版よりは2.7倍遅い

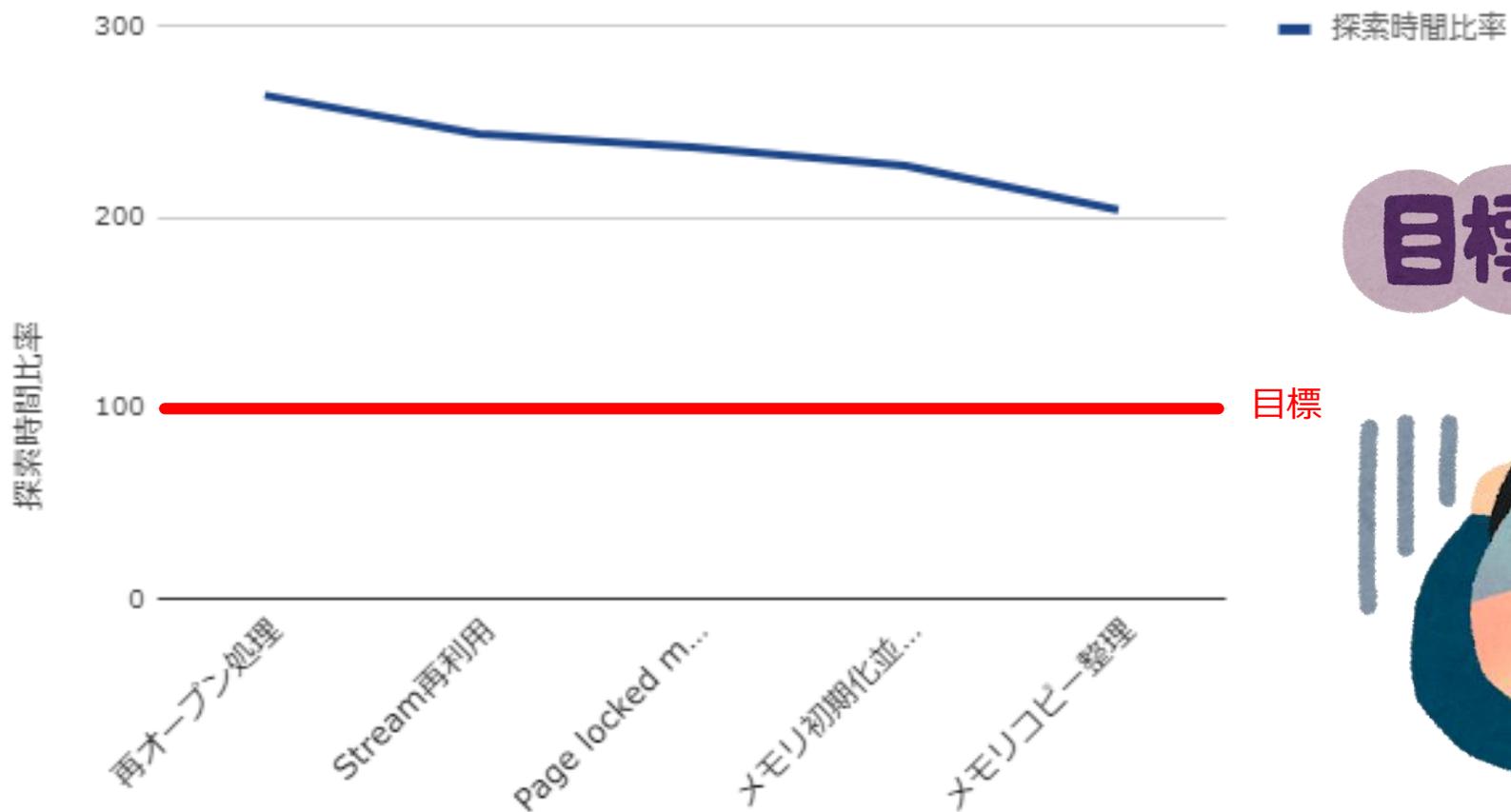
▶ 経路品質のため、ベルマンフォード法を採用しているが、ネットワークデータへの同時アクセスを防ぐ排他処理が足を引っ張っている

CUDAまわりで高速化



CUDAまわりで高速化

探索時間の推移 (青森→鹿児島)

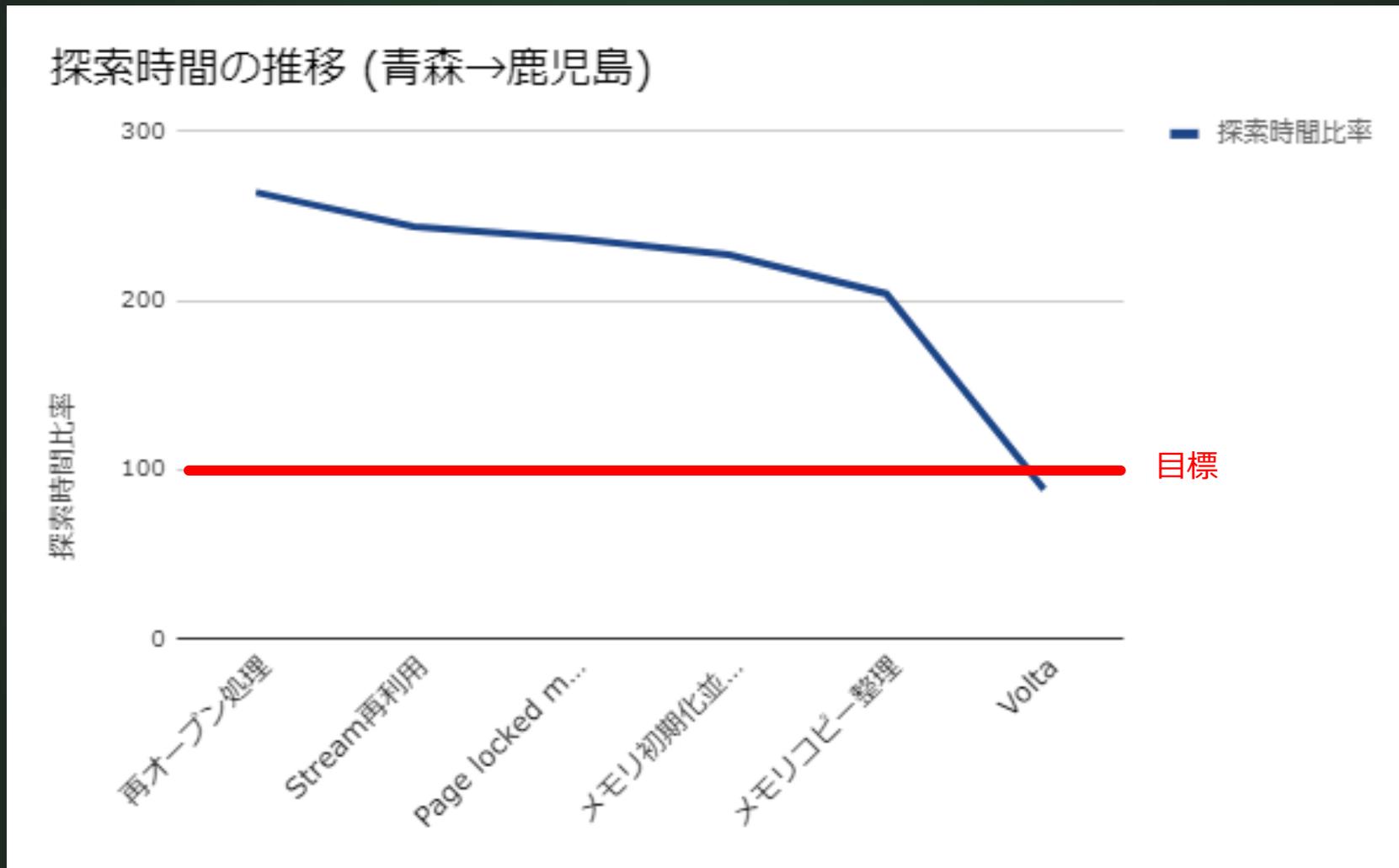


目標未達...

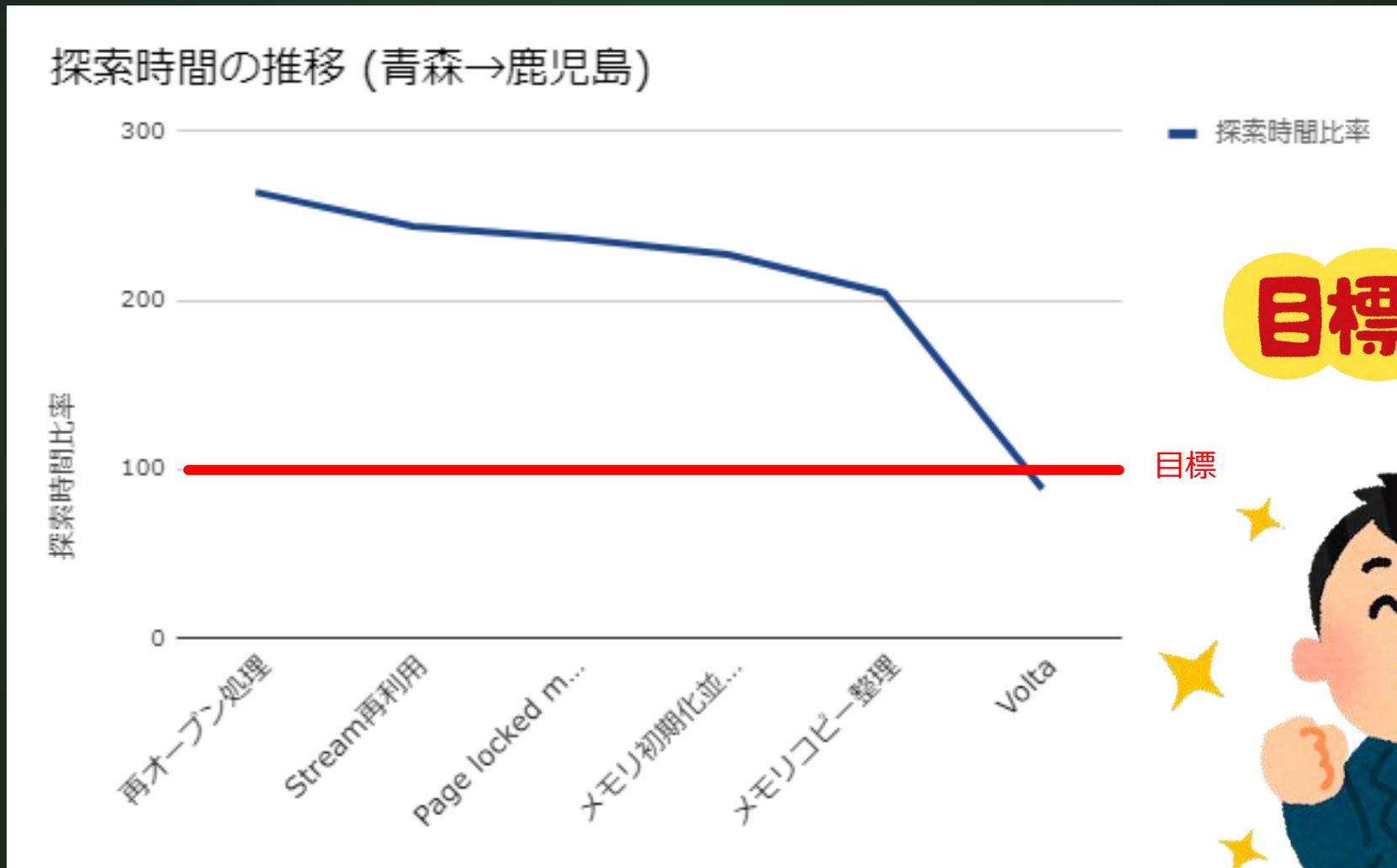


Voltaで実行するとどうなる？
p3.2xlarge

Volta(p3.2xlarge)で実行



Volta(p3.2xlarge)で実行

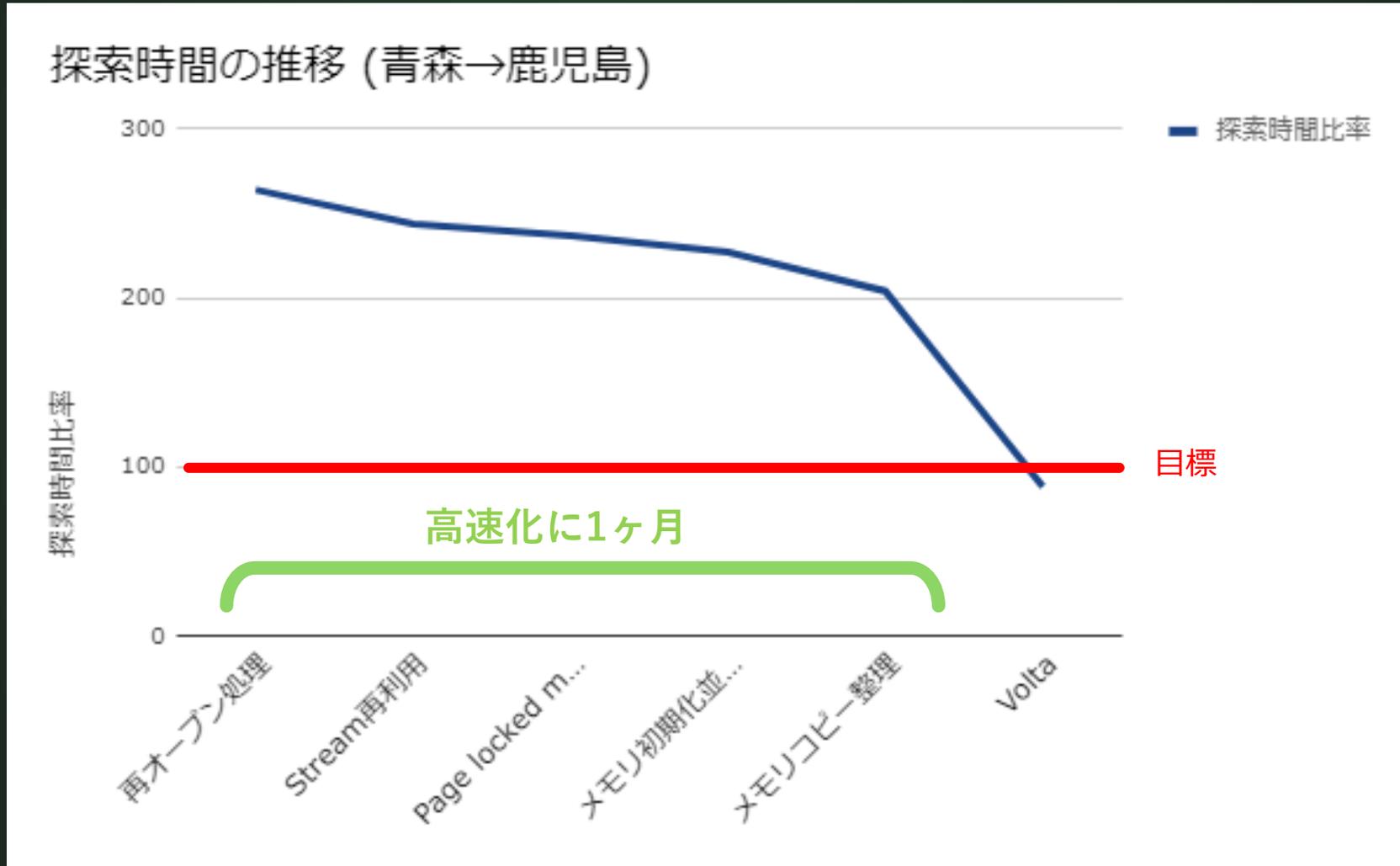


目標達成!

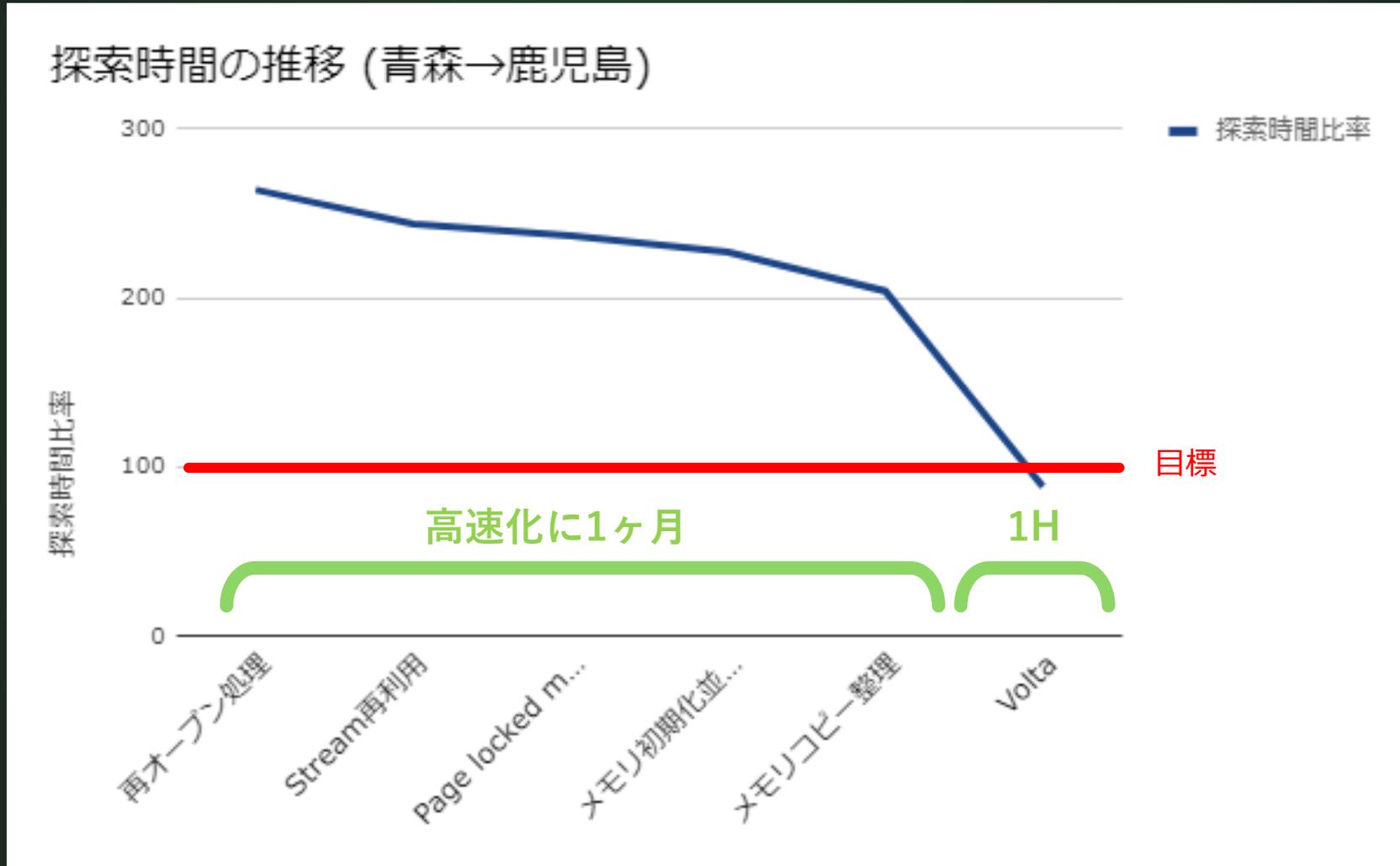


Voltaすごい

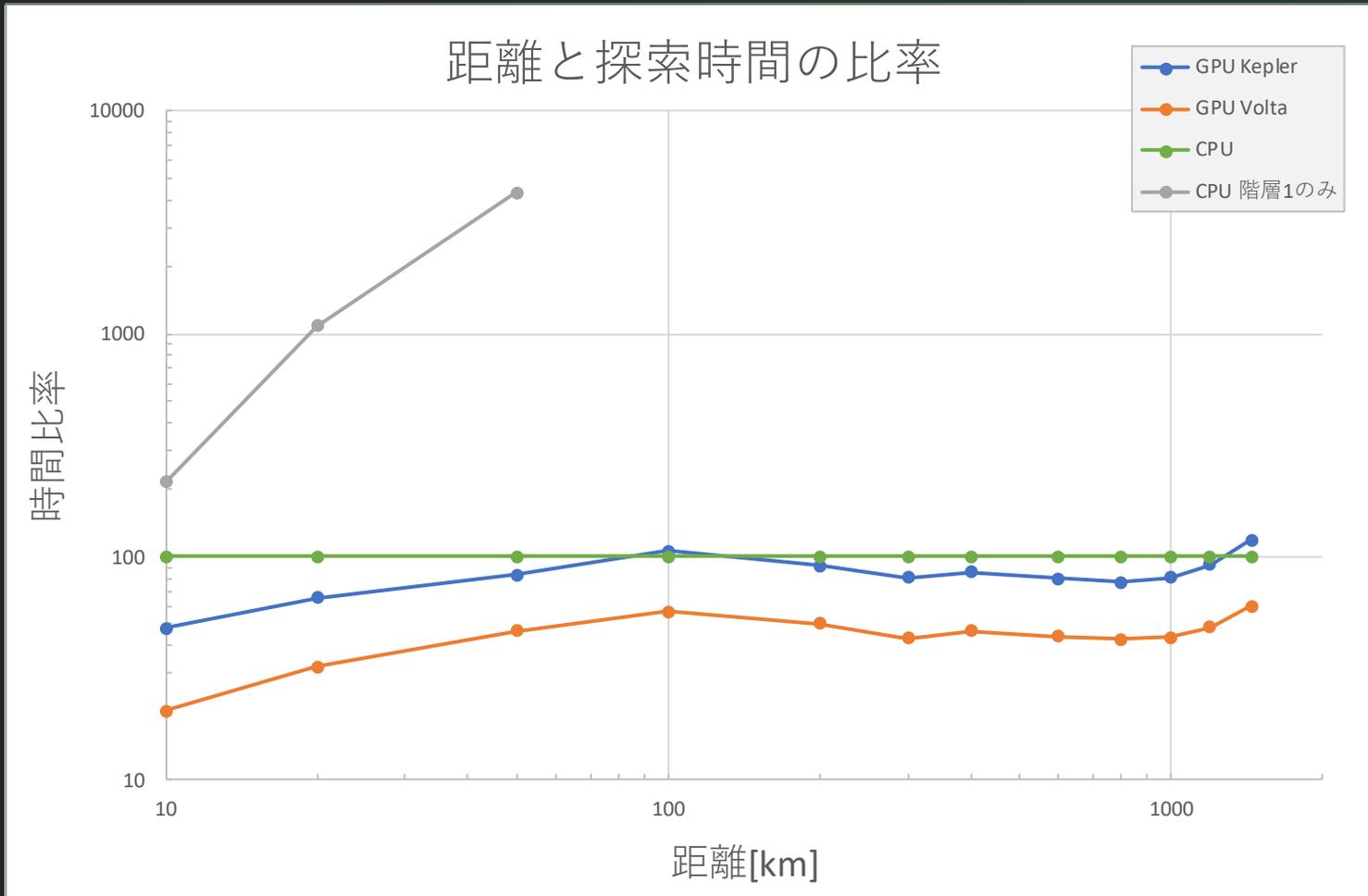
Voltaのすごさ



Voltaのすごさ



実行速度の最新状況

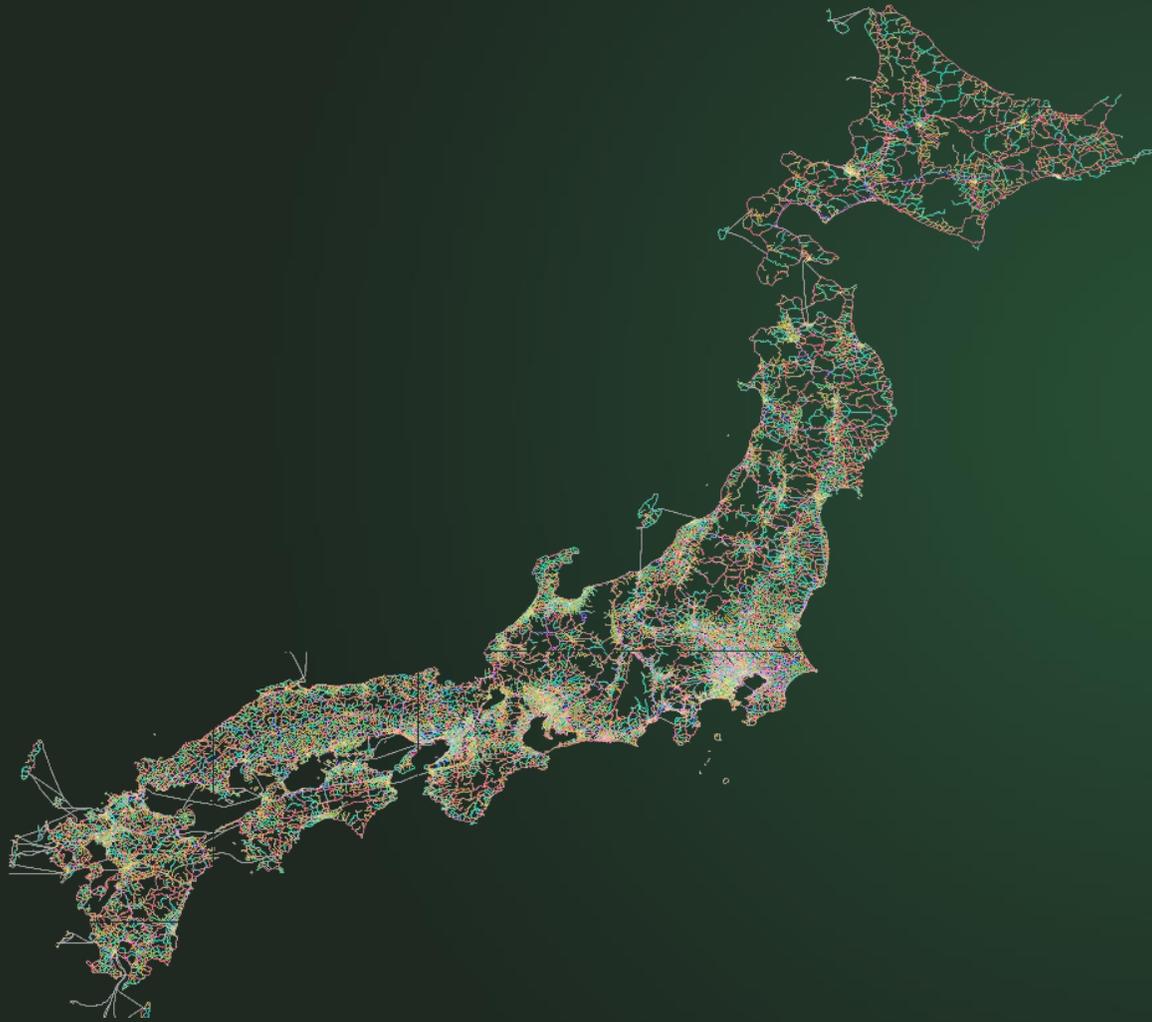


P2インスタンス
(Kepler)でもネット
ワークデータ省略版
CPUと同等まで高速
化

1000km以上の探索は
P3インスタンスへ割
り当てるようにすれ
ばOK

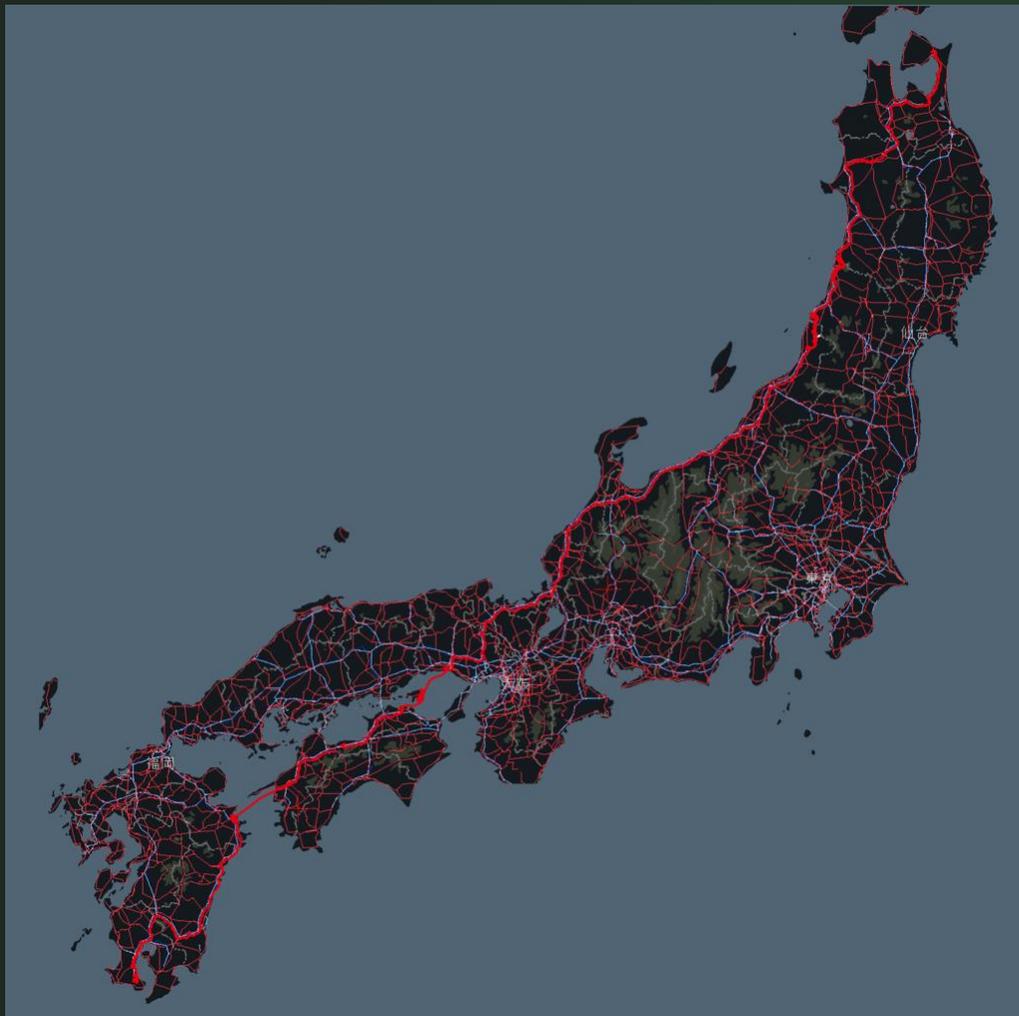
まとめ

チャレンジしたこと



GPUを活用した
経路探索エンジンを作る

結果



従来の方式よりも
より詳細なデータで
高速に探索ができた

結論

結論

- ▶ GPUでは不向きと言われている処理でもGPUを活用できた
 - ▶ 処理粒度が大きくても並列度が高いものには有効
- ▶ 近年のGPUの進化により、かなり汎用的な処理でも手軽に高速化できる
 - ▶ P2インスタンスで12GBのデバイスメモリがある
- ▶ AWS GPUインスタンスは低コストで手軽に試しやすく、インスタンスタイプを変えるだけで最新GPUを利用できる
 - ▶ サーバ用途としてはスポットインスタンスも魅力

GPUインスタンス
"機械学習以外でも"
どんどん使いましょう！！

まとめ

- ▶ マネージドサービスを使うことで、「**大事なことに集中**」できる
 - ▶ インフラエンジニア：アーキテクト
 - ▶ アプリケーションエンジニア：アプリケーション開発
 - ▶ . . .
- ▶ NAVITIMEではAWSのマネージドサービスをフル活用し、安定した先進的なサービスを引き続き提供

エンジニア募集中!!

- ・ サービス開発エンジニア
- ・ スマートフォン向けアプリ開発エンジニア (iOS/Android)
- ・ WEBエンジニア
- ・ アルゴリズム開発エンジニア
- ・ 地図・ナビゲーション開発エンジニア
- ・ システムエンジニア (エンジンライセンス)
- ・ ネットワーク・インフラエンジニア
- ・ デザイナー (UI/UX/web/アプリケーション/グラフィック)

自社開発でやってみたい方、ぜひご連絡ください。

<http://recruit.navitime.co.jp/recruit/career.html>



MAA



MINAMI AOYAMA NIGHT #6

2018.6.29 FRI 18:30-21:30

Presented by NAVITIME JAPAN

今回のテーマは「アンチパターン（失敗談）」

<https://minami-aoyama-night.connpass.com/>

NAVITIME