MITSUBISHI ELECTRIC
Changes for the Better

気づいてよかった。
アーキテクチャ設計後に色々と修正を余儀なくされたお話

2021/3/14

三菱電機インフォメーションシステムズ（MDIS）
技術部　野崎裕嗣

三菱電機インフォメーションシステムズ株式会社

MDIS

目次

- 当社の取り組み
  - 背景
  - ログ収集・分析基盤
- 見つかった問題と対応
  - 問題点①：Lambda関数のタイムアウト
  - 問題点②：Glueによる変換の利用コスト
  - 問題点③：Athenaによる分析の利用コスト
- 反省点と対策

背景

複数のサービス提供型新事業を立ち上げ

↓

利用者に安心感を与えるため、規格準拠に取り組む

↓

規格に対応する監査機能を開発

ログ収集・分析基盤

ログ収集・分析基盤

①概要

- 監査やインシデント対応の効率化を目的に、
  サービスごとに分散したログ情報を収集して、
  一か所で分析する共通基盤。

ログ情報

- サインイン、操作
- セキュリティ製品
- サービス自体

分析（例）

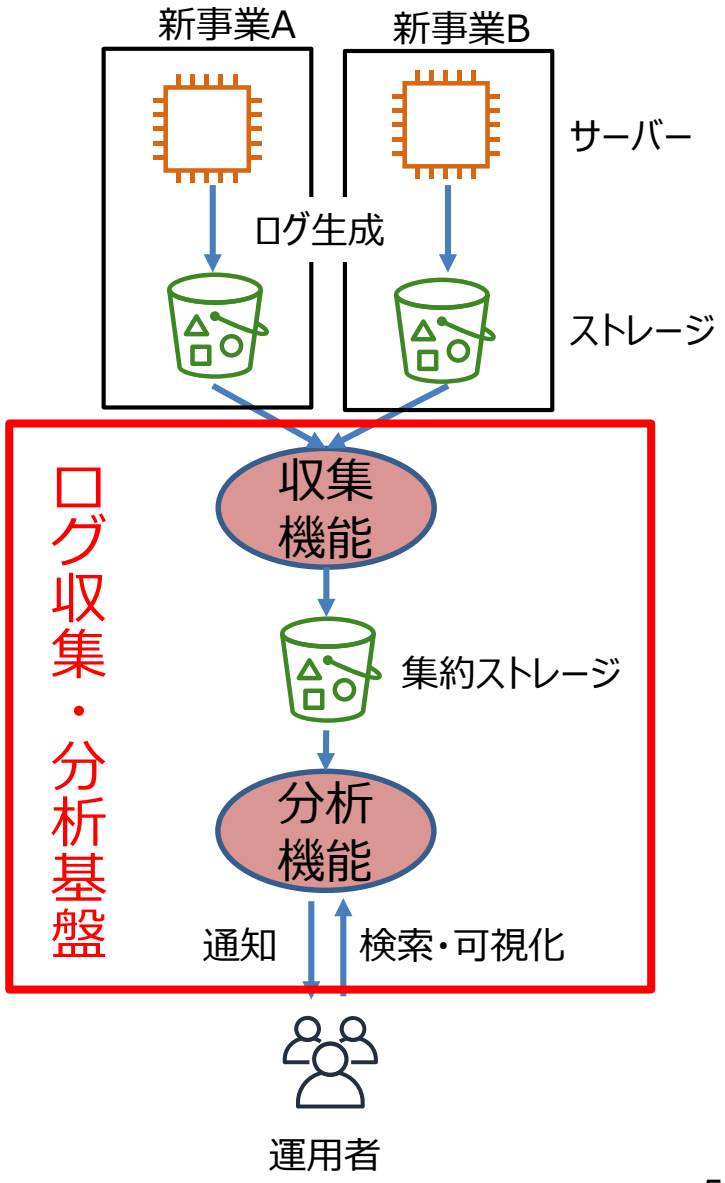
✓ 障害発生時の原因究明
✓ 不正ログインの点検
✓ サービスのアクセス数集計

ログ収集・分析基盤

②提供機能

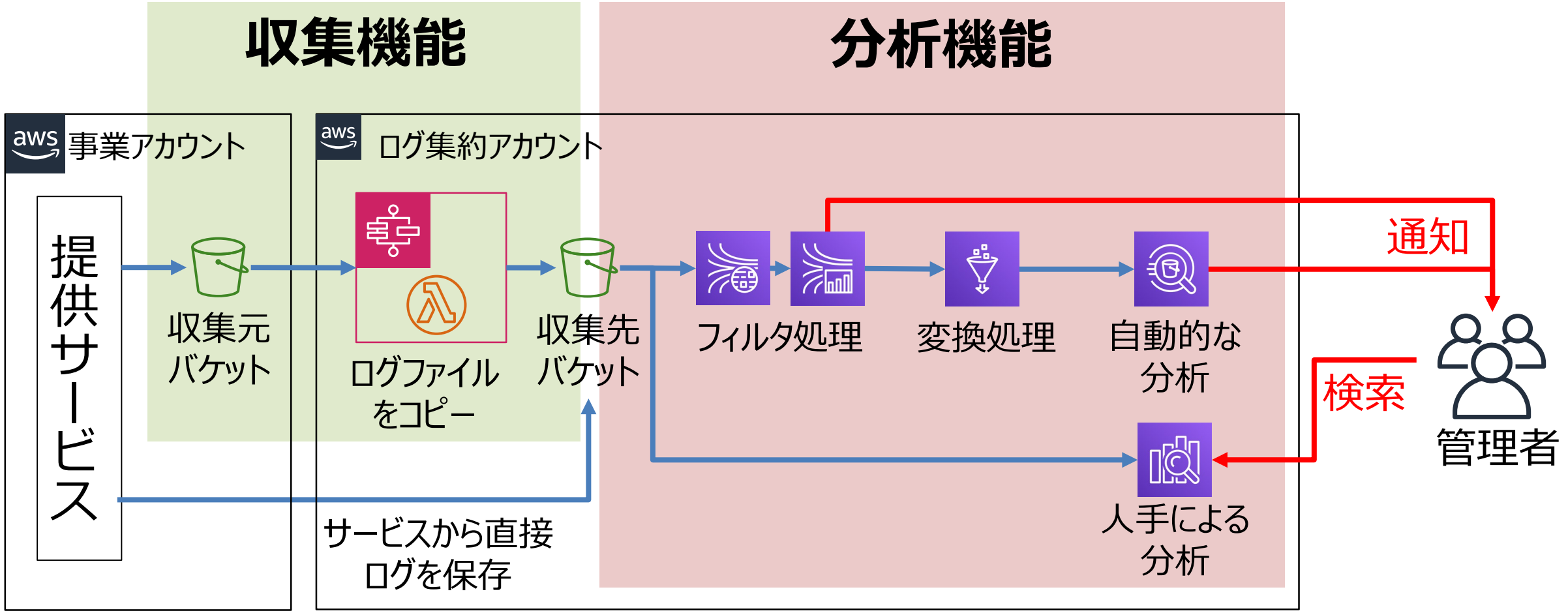
すべてのログを**同じ場所**から**同じ方法**で分析

- 収集機能
  - 各新事業が出力するログを1ヵ所に集約
- 分析機能
  - 特定のログを検出して**通知**
  - 人手によりログを任意に**検索**
  - ログ情報を集計して**可視化**



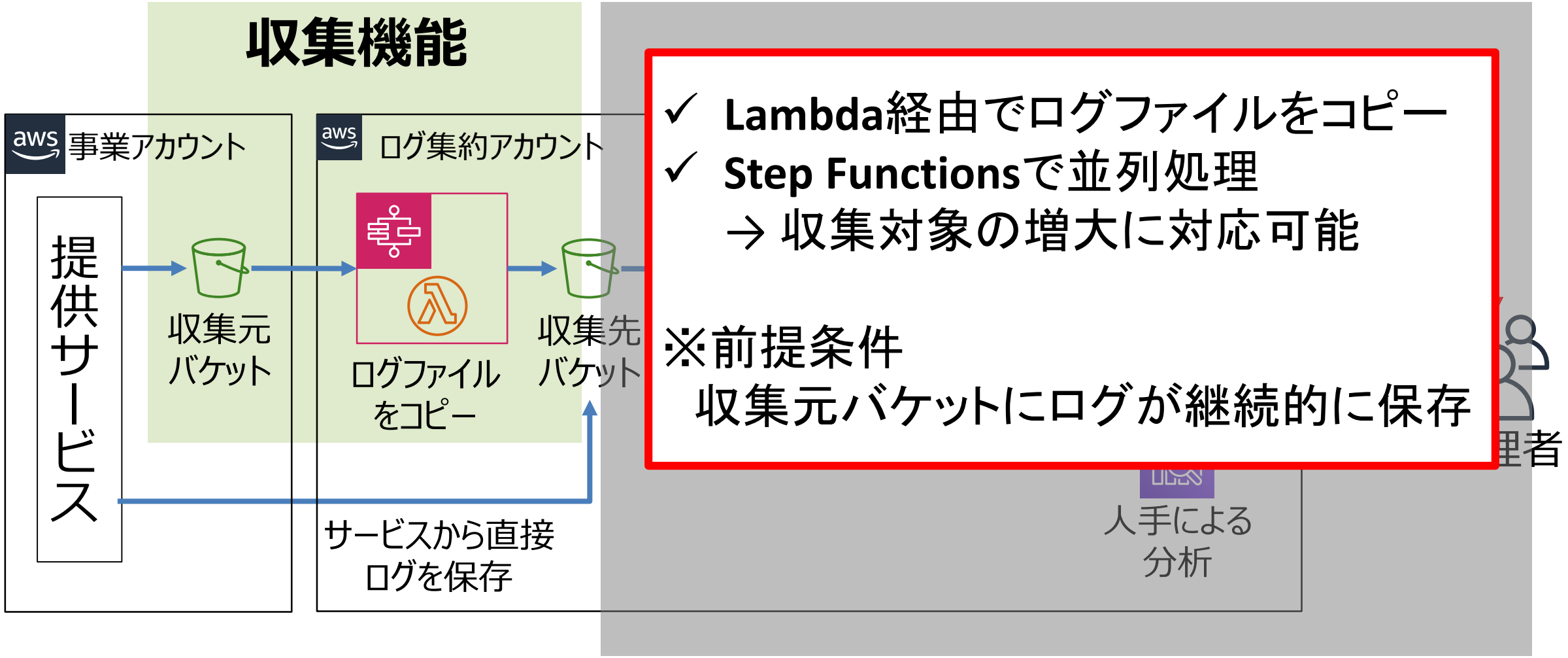
ログ収集・分析基盤

③実現方法



ログ収集・分析基盤

③実現方法（収集機能）

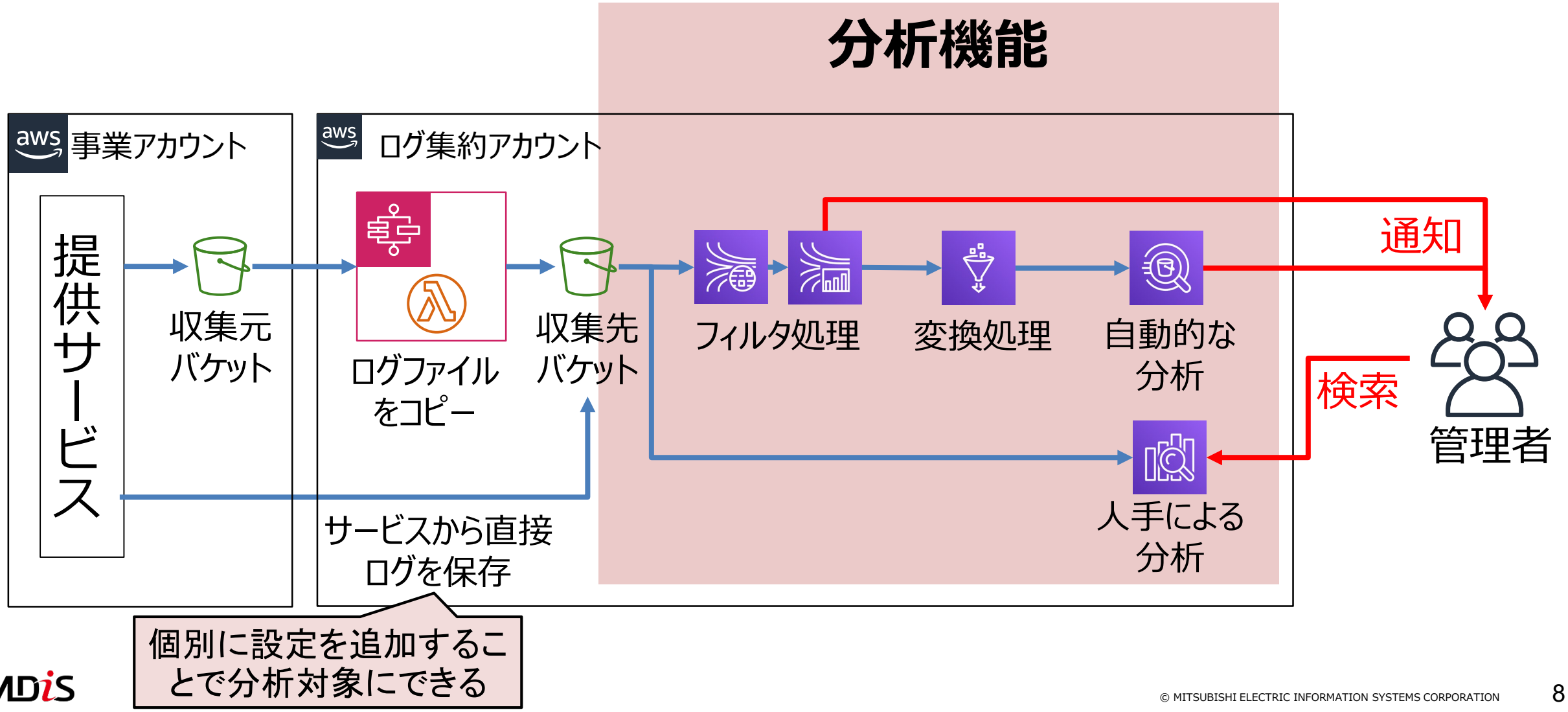


- ✓ **Lambda経由でログファイルをコピー**
- ✓ **Step Functionsで並列処理**
  → 収集対象の増大に対応可能

※前提条件
収集元バケットにログが継続的に保存

ログ収集・分析基盤

③実現方法（分析機能）



ログ収集・分析基盤

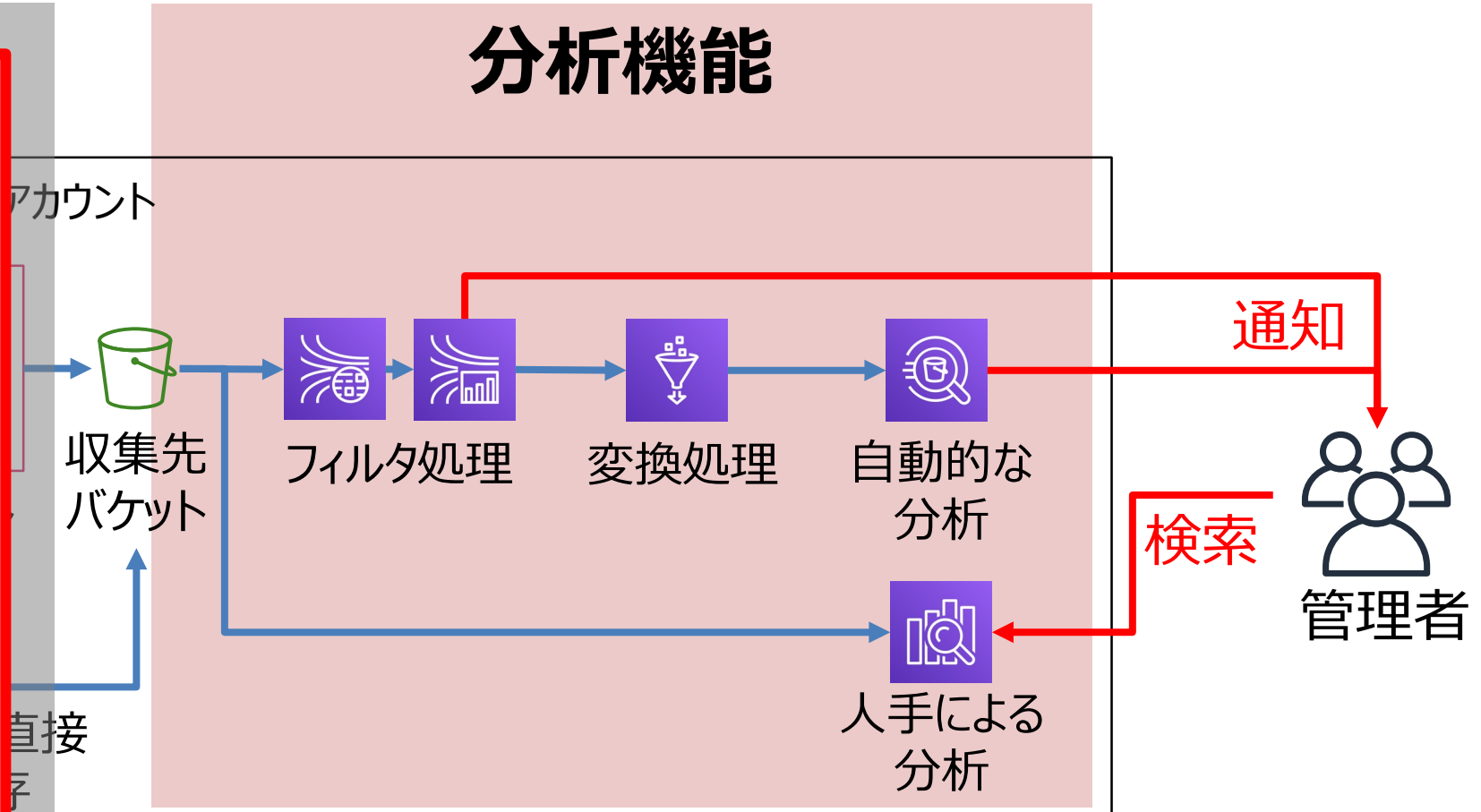
③実現方法（分析機能）

① Kinesisでフィルタして通知

② Glueでログを変換

③ Athenaで自動分析して結果通知

④ Elasticsearchで追加の分析や可視化



ログ収集・分析基盤

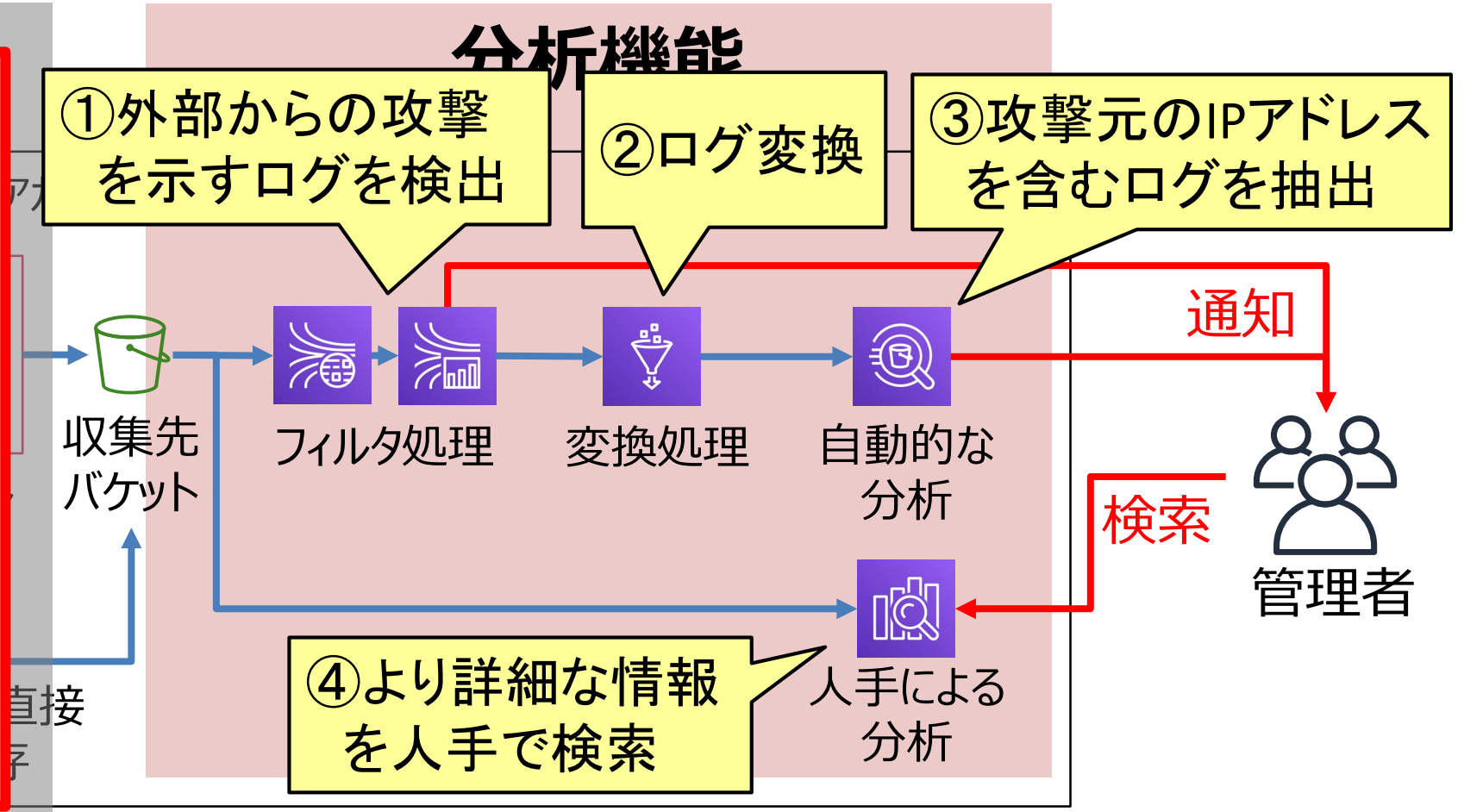
③実現方法（分析機能）

① Kinesisでフィルタして通知

② Glueでログを変換

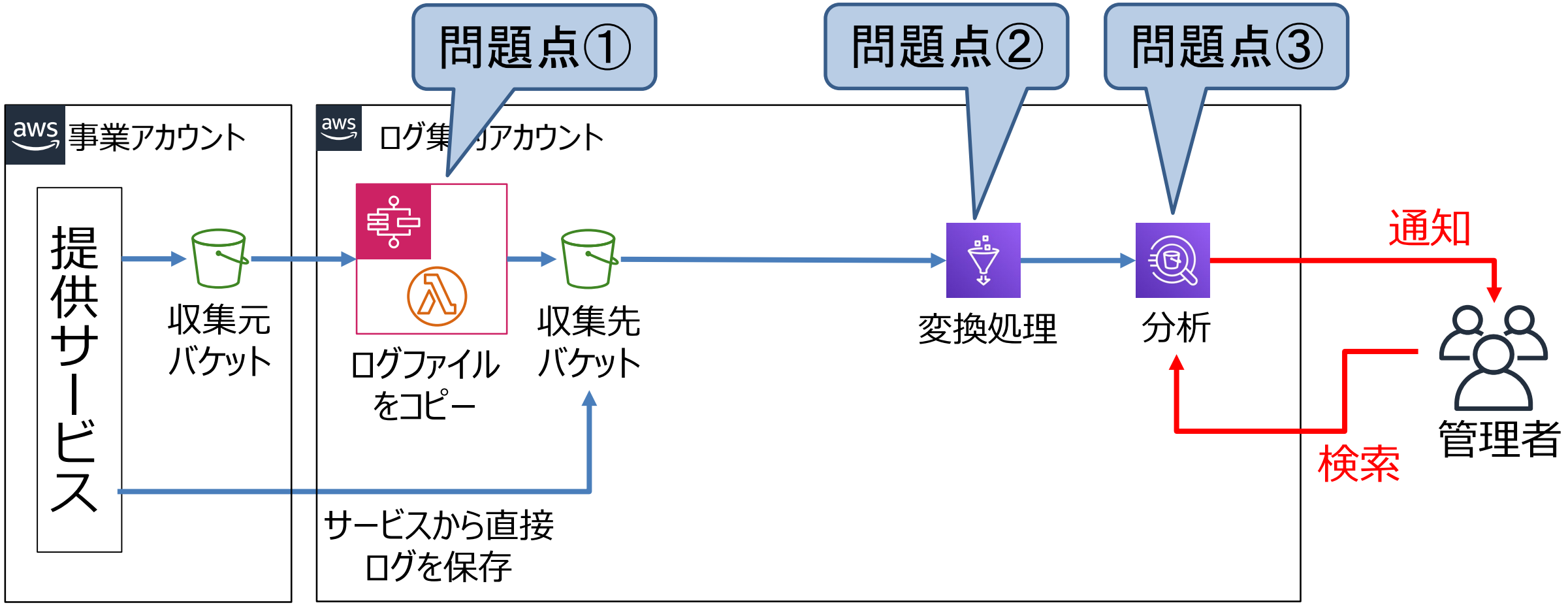
③ Athenaで自動分析して結果通知

④ Elasticsearchで追加の分析や可視化



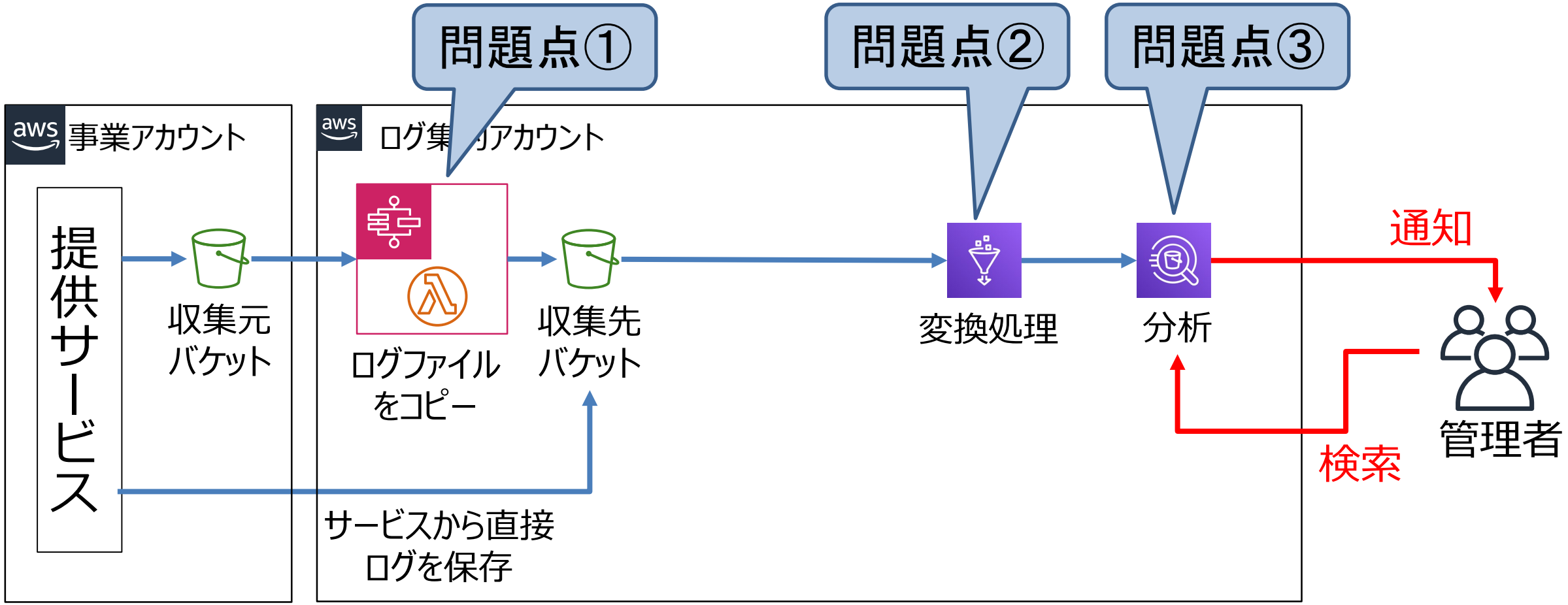
設計後に見つかった問題点

設計直後の構成図



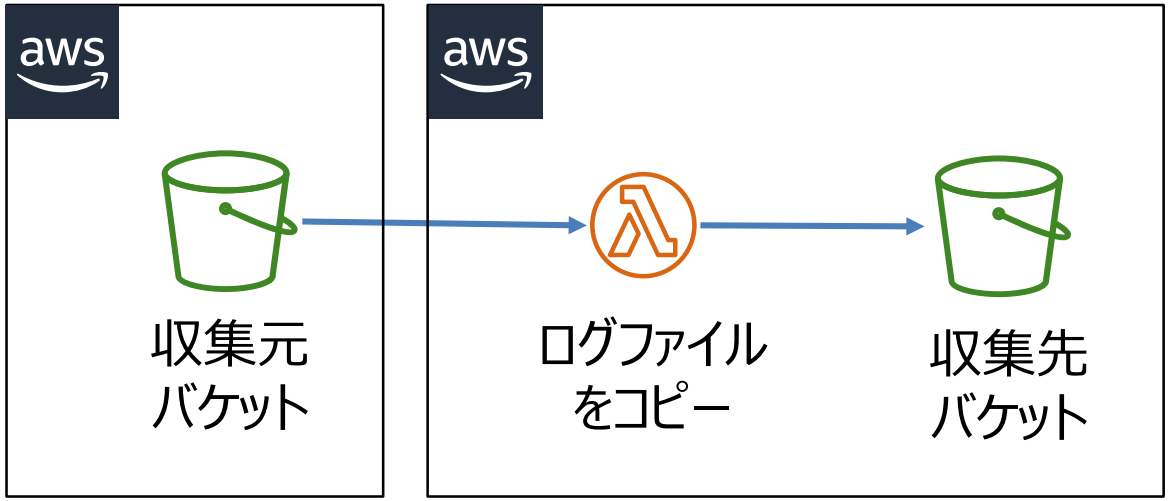
問題点①：Lambda関数タイムアウト

- 大容量ファイルコピー時のLambda関数のタイムアウト



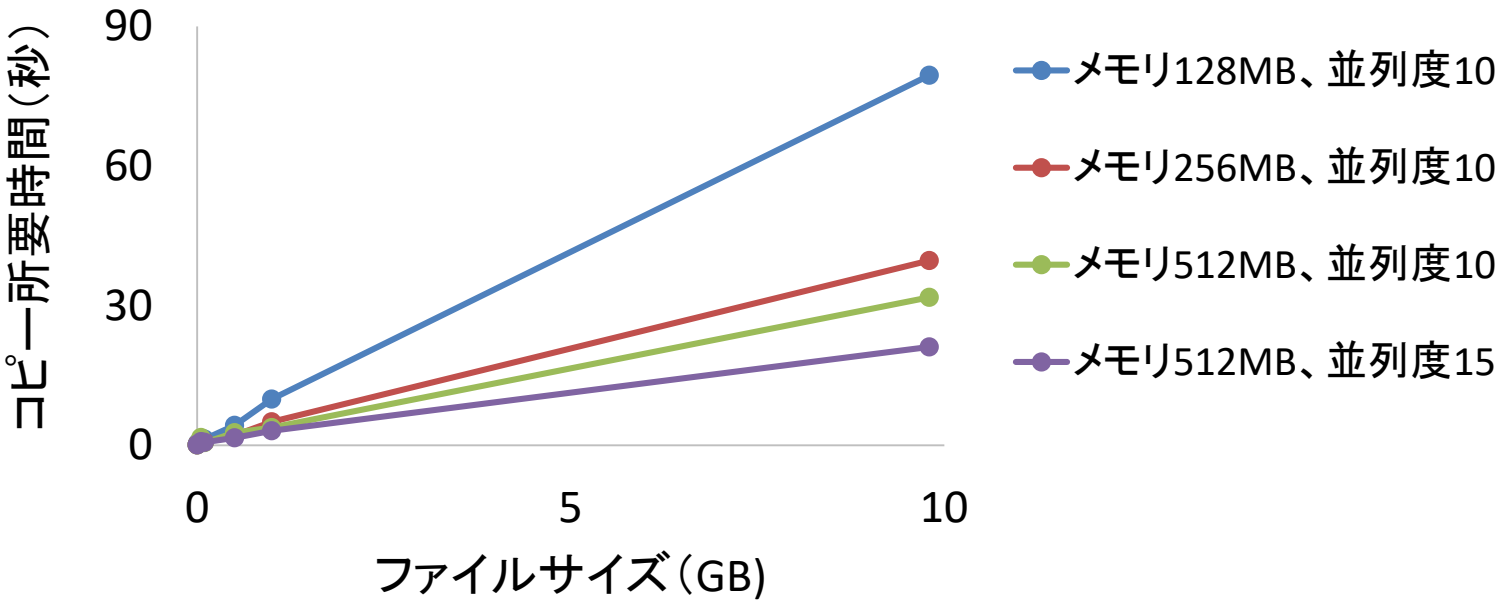
問題点①：Lambda関数タイムアウト

- 経緯
  - 管理コスト削減のため、EC2ではなくLambdaを採用
  - Lambda関数の最大実行時間：15分
  - S3のファイルコピーは同期処理



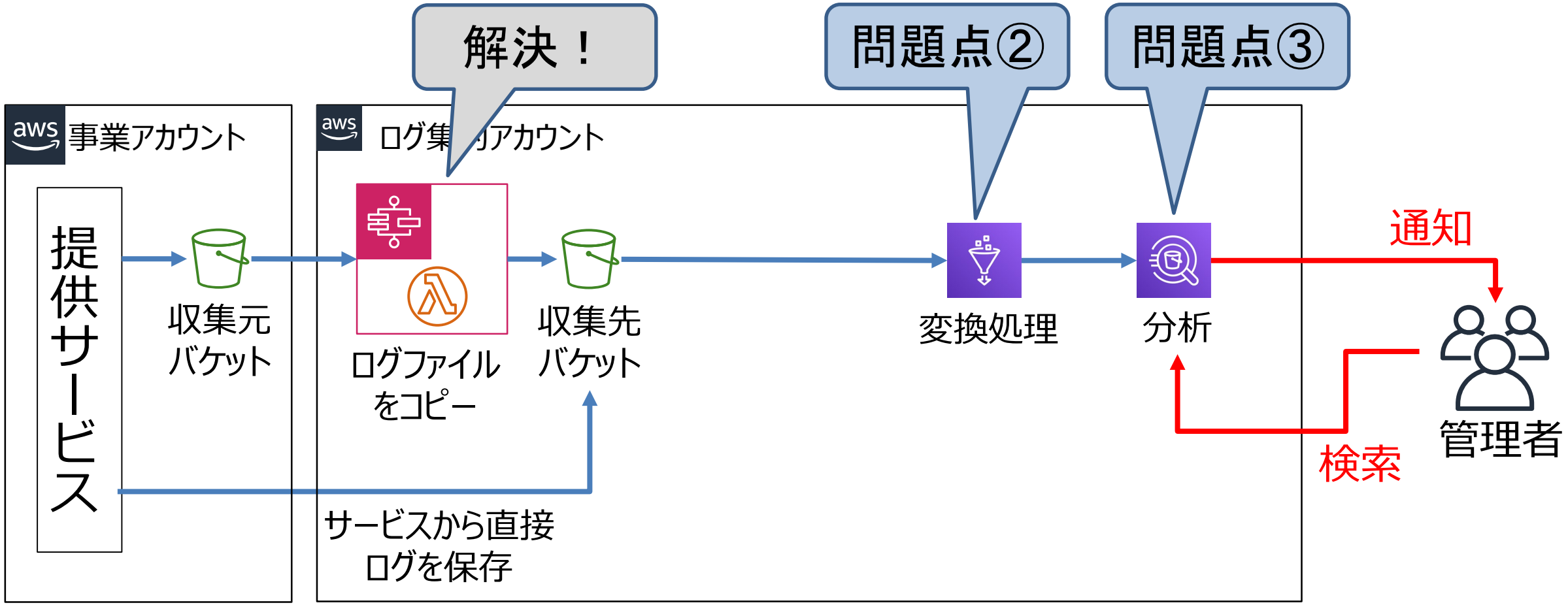
問題点①：Lambda関数タイムアウト

- 対処方法
  - Step Functionsを活用して処理を分割
  - 想定ログサイズを試算して性能測定
    → 10GB でも 90秒未満 (上限の10%)
  - 設定変更でコピー所要時間を短縮できることを確認



問題点②：Glue利用コスト

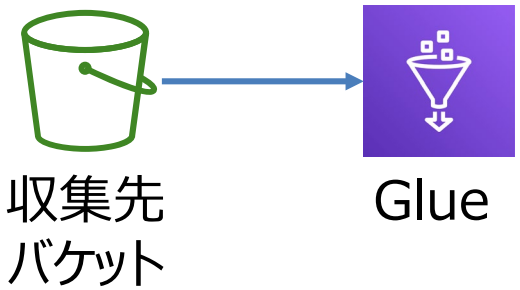
- Glueですべてのログを逐次変換するとコストが増大



問題点②：Glue利用コスト

- 経緯
  - 過去事例に基づき、変換処理にGlueを採用
  - ログ追加をトリガーにGlueを呼び出すと実質 常時稼働の状態
  - 費用は 0.015 USD/分 (最小構成 DPU:2の場合)
    → 1ヵ月動かし続けると 約650 USD
  - 少量のログだとリソースが過剰

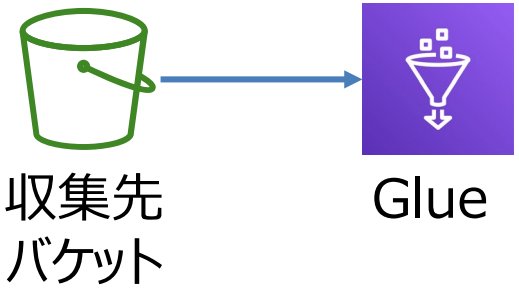
すべてのログを変換



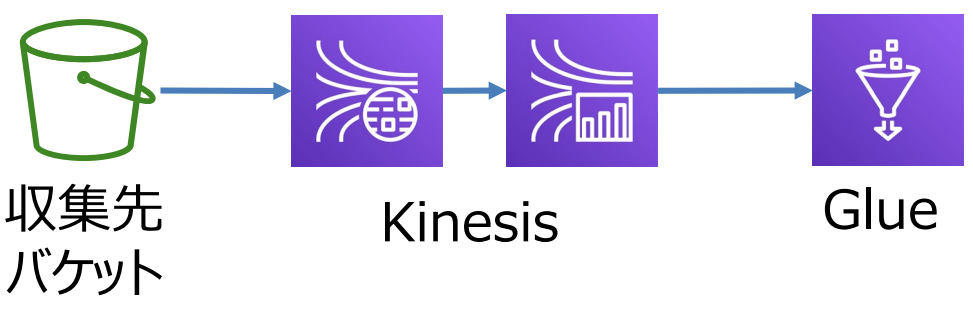
問題点②：Glue利用コスト

- 対処方法
  - すべてのログは一度Kinesisに投入してフィルタ
    →追加の分析が必要な場合のみGlueで変換を行う
  - 短時間に連続する場合はデータを退避してスキップ

すべてのログを変換

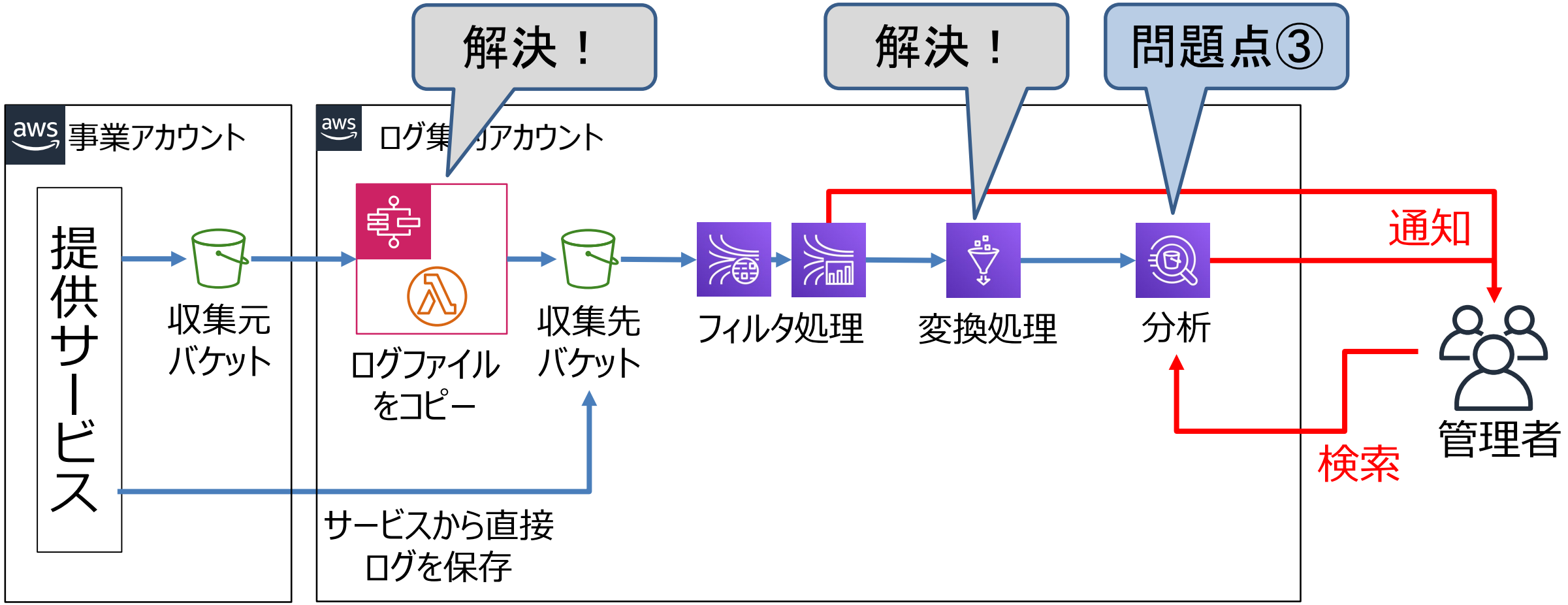


フィルタして必要な場合のみ変換



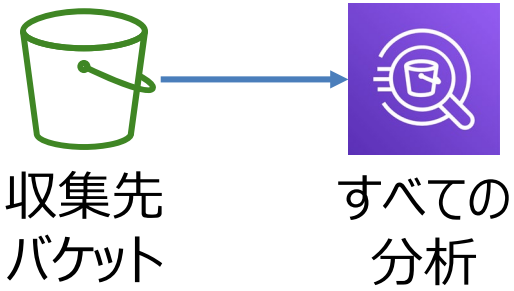
問題点③：Athena利用コスト

- Athenaで何度も繰り返し検索を行うとコストが増大



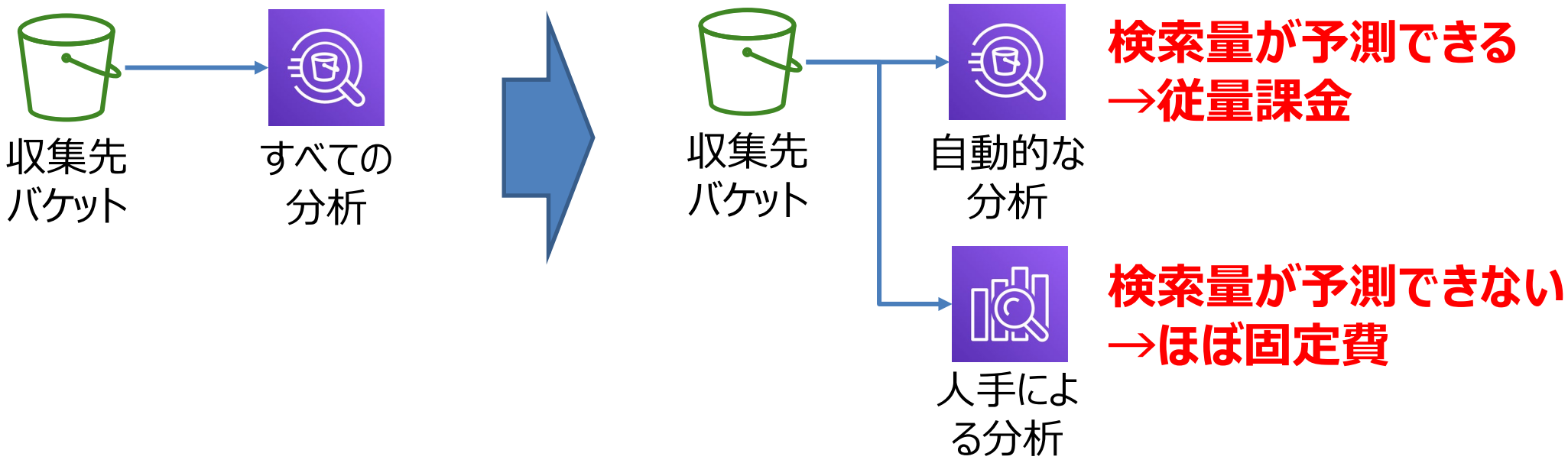
問題点③：Athena利用コスト

- 経緯
  - 過去事例に基づき、分析処理にAthenaを採用
  - 対象のデータ 1TB あたり 5 USD
    → 100GB(0.1TB)を1000回検索すると500 USD
  - 実際にどの程度分析を行うか予測が難しい
    → 障害の調査時にコストを心配したくない



問題点③：Athena利用コスト

- 対処方法
  - 人手による分析は、ほぼ固定費で利用できるElasticsearch
  - 今後の拡張性を考慮して、Athenaは残す
    → 検索量が予測できる自動的な分析にはAthenaを利用



反省点と対策

- 反省点（なぜ起きたか？）
  1. 過去事例を参考にするときに検討が不足
  2. データフローだけに着目して、頻度や量の考慮が不足
  3. 実際の利用を想定した細かいコスト評価の不足

- 問題を通じて得られた知見
  1. Lambdaは、設計時にタイムアウト制約の検証が必要
  2. Athenaは、検索量が不明瞭の場合には注意が必要
  3. Glueは、バッチ処理には適しているが、逐次処理には不向き

反省点と対策

- 今後の対策（まとめ）

1. ベストプラクティスや過去事例に似たような構成があっても、
   そのまま鵜呑みにしない。
   - ✓ 要件や使い方の小さな違いで、適さなくなるケースもある。

2. 性能やコストの観点での評価もしてから採用する。
   - ✓ 同じことを実現したくても、使い方によって適切なサービスは変わってくる。
   - ✓ 1つに固執せず、複数の選択肢を用意できる必要がある。

3. サービスの制限事項をナレッジとして蓄積し、共有する。
   - ✓ 1人では制限や仕様を見落としてしまうこともある。
   - ✓ 集団で知見を集めて、気軽に情報交換できる環境が必要。

ご清聴ありがとうございました

ご注意

・本書の内容の一部又は全部を当社に断りなく、いかなる形でも転載又は複製することは、固くお断りします。

・本文記載の社名、製品名、ロゴは各社の商標または登録商標です。