

# メールを用いたエキスパート 検索



# 背景

---

- 大規模なソフトウェア開発現場（企業）では多数の開発者が開発に関わっている
- 企業全体でも複数のプロジェクトが同時に進行している
- そのため、誰が過去・現在にどういった仕事に関わったかを把握することは難しい

# 問題と目的

---

## ◎ 問題

- ▶ 適任の人物をプロジェクトに割り当てられない可能性があるため、プロジェクトの効率が悪い
- ▶ 企業が案件を受けるときに、適任の人物がいるにも関わらず案件を逃してしまう

## ◎ 目的

- ▶ 過去・現在に誰がどういったプロジェクトに関わりがあるのかを把握できるようにする

# 既存研究

---

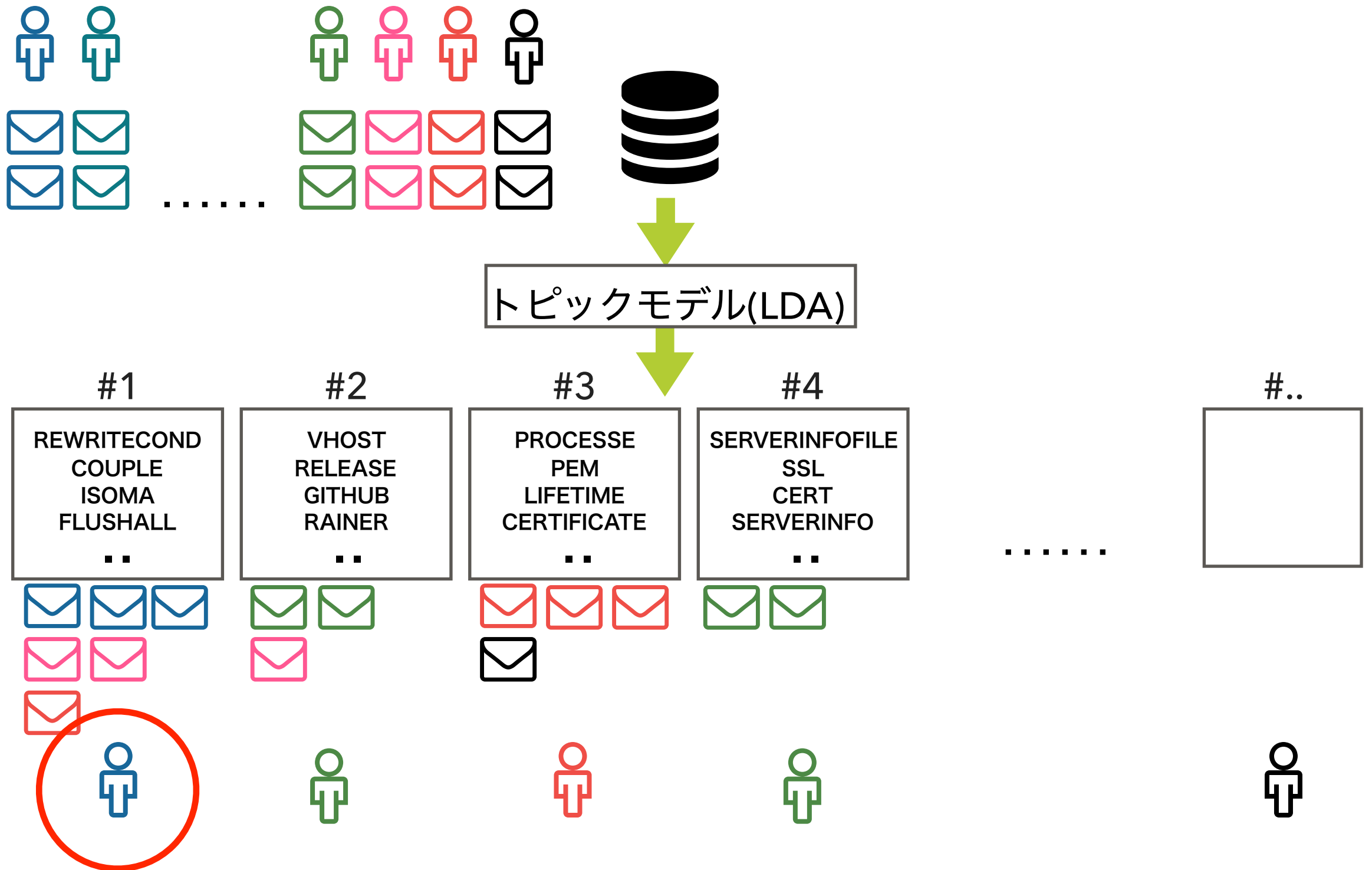
- ◎ ソースコードの変更履歴から、各モジュールに精通している開発者を探し出す
  - ▶ 狭い範囲・ソースコードのみ
- ◎ 問題点
  - ▶ ソースコードを変更している開発者が対象
  - ▶ ソフトウェア開発はソースコードを書くだけではない
  - ▶ 設計を行うほうがより高度な知識が必要

# アプローチ

---

- 全ての開発者が使用するであろうメールを分析する
  - LDAを用いて分類する
    - ▶ LDAでは、直接的に一緒に使われていない単語であっても同一トピックに分類することができるため
1. 全メール(添付ファイルを含む)に対して、前処理を行う
  2. 名詞だけでLDAを用いてメールをトピックごとに分類（トピック数は適当）
  3. トピックごとにメールの数に応じて、開発者をスコアリング

# アプローチ



# 現在のアプローチの問題

- パラメータの設定
  - ▶ トピック数を決める必要がある
- トピックの構成語だけでは, 意味が分かりにくい
- トピックが分散している可能性がある

・トピックと構成語の例

| 0         | 1           | 2          | 3       | 4          |
|-----------|-------------|------------|---------|------------|
| september | ssl         | sandbox    | afaic   | scheme     |
| provider  | certificate | grep       | freitag | intent     |
| jump      | tenc        | redirect   | rainer  | community  |
| errorlog  | referral    | convert    | sat     | compressio |
| oct       | referral    | appreciate | sign    | rewritecon |
| vhost     | decide      | acces      | perl    | vendor     |