



Dev-07: Developer Productivity

『リアルタイムターゲティング広告の 配信・分析基盤の開発の裏側』

～ Amazon Kinesisの活用 ～

株式会社サイバーエージェント
アドテクスタジオ Dynalyst
木村 衆平

Profile

名前：

木村 衆平 Shuheii Kimura

<https://www.facebook.com/shuheii.kimura.10>



所属：

株式会社サイバーエージェント
アドテク本部 AdTechStudio
Dynalyst 開発責任者

経歴：

- 2011/04 ~ 広告効果計測システムの開発
- 2011/11 ~ PCのRTB広告配信における最適化エンジンの設計/開発
- 2012/09 ~ SP向け海外DSPシステムの設計/開発
- 2013/11 ~ SP向けダイナミックリターゲティング広告配信システムの設計/開発

Agenda of this session

オンライン広告のいま

- 昨今の広告配信に求められることは何か？
- Dynalystが提供する価値とは何か？
- Dynalystのシステム構築におけるコンセプト

システムの大規模化に伴って直面した課題たち

- ログの量だけでなくログの内容が多様化することに対応しにくい問題
- データの利用用途の多様化と柔軟性の問題
- レポート集計基盤 ≠ アドホック分析基盤

Amazon Kinesisの採用とその導入の過程

- なぜKinesisで解決できると考えたか？
- Kinesisをコアとするデータ活用戦略と実現したシステムについて

Significant changes of the environment for online display advertising

～ オンラインディスプレイ広告を取り巻く環境 ～

What's happening ?

より細かなターゲティングが可能な時代

- 5W1HにおけるWhoが特に重要視される
- ユーザの行動に基づく広告配信
- 広告予算の適切な配分とROI効率の向上

Are you an user whom
we should deliver our ad ?



広告出稿の意思決定はリアルタイム化 ~ 50msec or die ~

- いつ, どこで, だれに, どんな広告をいくらで出す or 出さない
- Real Time Bidding 取引は当たり前の世界

広告配信技術は年々、高度に複雑に...

What is ..

 Dynalyst

RealTime User Personalized Dynamic Ad service Provider

ユーザーのサイト内行動をベースに、エンゲージメントレベルを解析。
見込み顧客かどうかをリアルタイムに判別し、ターゲティング広告を展開します。



 **Dynalyst**

is fully powered by



Ad1 Tech
STUDIO

Keywords

RealTime ~ 何がリアルタイムなのか? ~

- サイト内ユーザ行動がリアルタイムに広告に反映される
- RTB時代だからこそできる

User Personalized Ad ~ レコメンド技術 ~

- ユーザの趣味嗜好に合わせて広告がダイナミックに変化



Key Concept for Dynalyst System Architecture

システム運用コストは小さく

~ チームバランスを最適化、Ops + Dev + Biz ~

少し先を見越したスケーラビリティ

~ スピードを優先して意図的に無視して負債を作るケースはもちろんあります ~

作らなくて済むものは作らない

~ そもそも作らなければメンテしなくてよい ~

~ AWSはS.O.Aなのでソリューションは探す、もしくは相談すれば見つかる ~

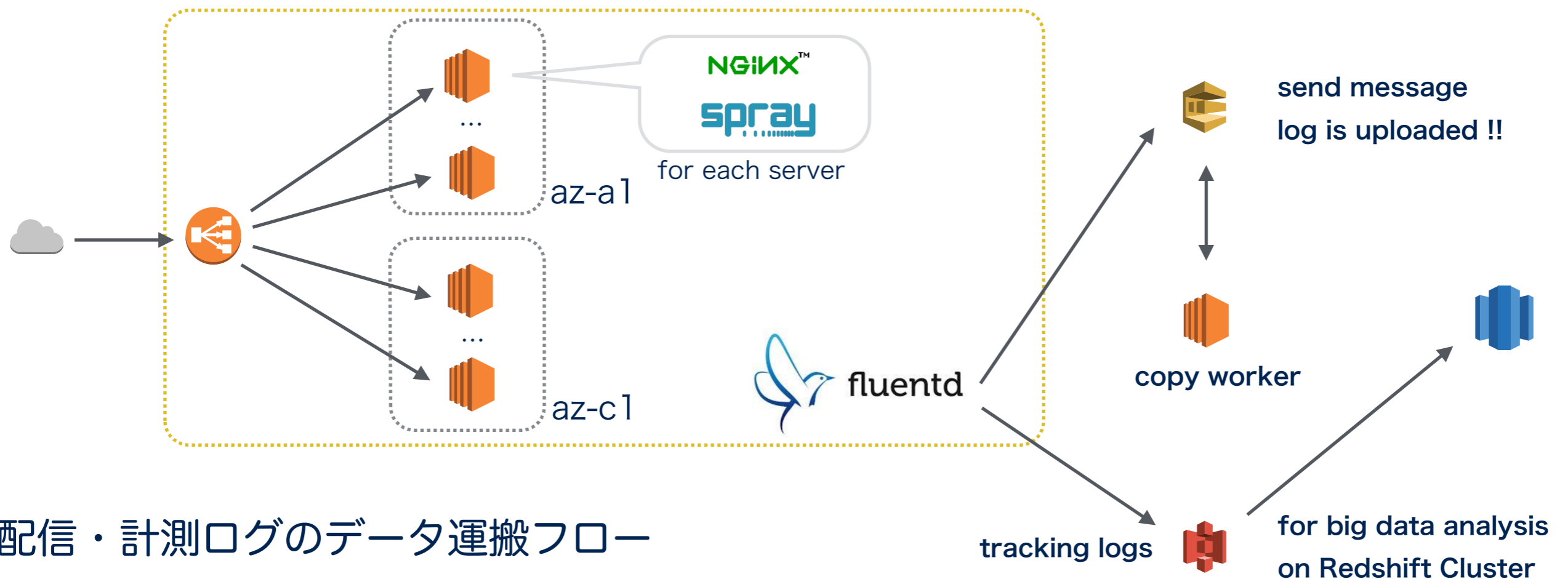
Problems we faced

when business & system expands

～ システムの大規模化に伴って直面した課題たち ～

Data volume & Data variation

データの量への配慮 ○ データの質への配慮 △



配信・計測ログのデータ運搬フロー

① fluentd driven data operation

② S3 => Redshift の継続COPY過程で**テーブル定義を容易に変更できない**

※ 『あの情報も欲しいなあ。』 ⇒ **カラムを追加!?**

Different query requirement from same data source

頑張ればRedshiftでなんでもできた立ち上げ期！ ⇒ そろそろ見直す時期？

配信実績分析のためのクエリ

```
1 SELECT
2   advertiser_id,
3   publisher_id,
4   creative_size_id,
5   COUNT(distinct imp_id) as imp_count
6 FROM
7   impression
8 WHERE
9   imp_time > '2015-05-01 00:00:00' AND
10  imp_time < '2015-05-01 01:00:00'
11 GROUP BY
12  advertiser_id,
13  publisher_id,
14  creative_size_id
```

 access logs

広告主様向けのレポート集計クエリ

```
1 SELECT
2   advertiser_id,
3   COUNT(distinct imp_id) as imp_count
4 FROM
5   impression
6 WHERE
7   imp_time > '2015-05-01 00:00:00' AND
8   imp_time < '2015-05-01 01:00:00'
9 GROUP BY
10  advertiser_id
```

 fluentd

 copy worker

 trac



for big data analysis
on Redshift Cluster

アドホッククエリと業務系クエリが同居しているのはそろそろ限界 !?

※ WLMなどでのクエリ優先度調整, 中間テーブル設計などではどうにもならない精神的負担

Fresh Data ? or Big Data ?

データの**鮮度**が何よりも重要？

過去の配信実績の傾向などデータの**量**が重要？

ある広告枠の在庫及び単価はどうか？

ある広告枠の配信費用効果はどうか？

どんな商品の広告をどれだけ配信したか？

ある商品の広告を配信しすぎていないか？



Redshiftまでデータを運んでからがスタートではちょっと遅い!?

How we solved these problems
by using Amazon Kinesis ..

~ **Kinesisの採用とその導入の過程** ~

Why Amazon Kinesis ?



Concept Matches

1:input => n: output **IMPORTANT**

- Kinesis Streamを起点としたデータフローが要件に合いそう??



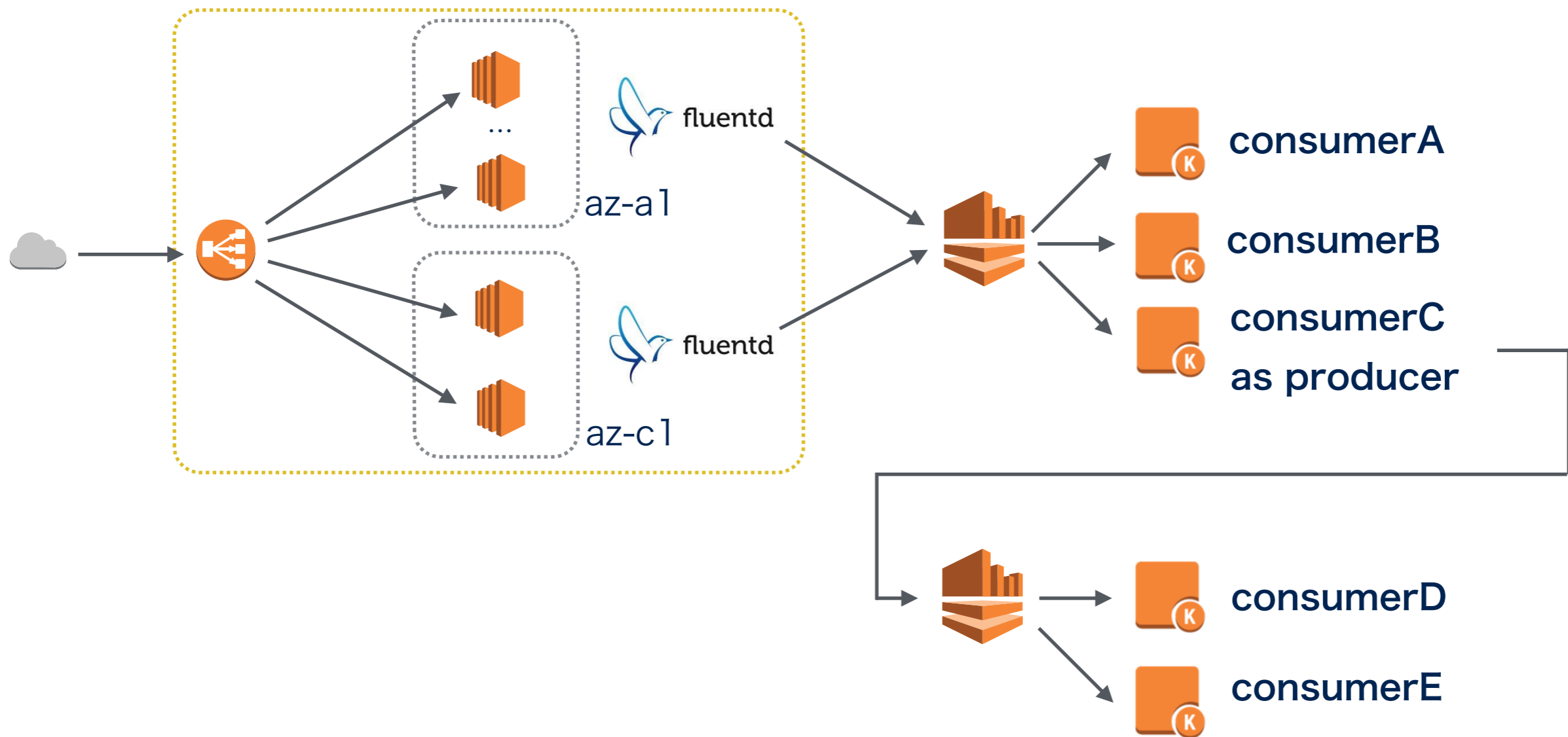
Fresh Data & Big Data **IMPORTANT**

- S3まで運べればRedshiftとのIntegrationはできる??
- その場でデータを取り出して処理すればFreshデータとして扱える??

分析基盤と集計基盤の役割の分離

- データフローをシンプルにしつつ分析基盤強化のための施策を打ちたい
- S.O.Aの思想そのままに色々なサービスと連携できる??

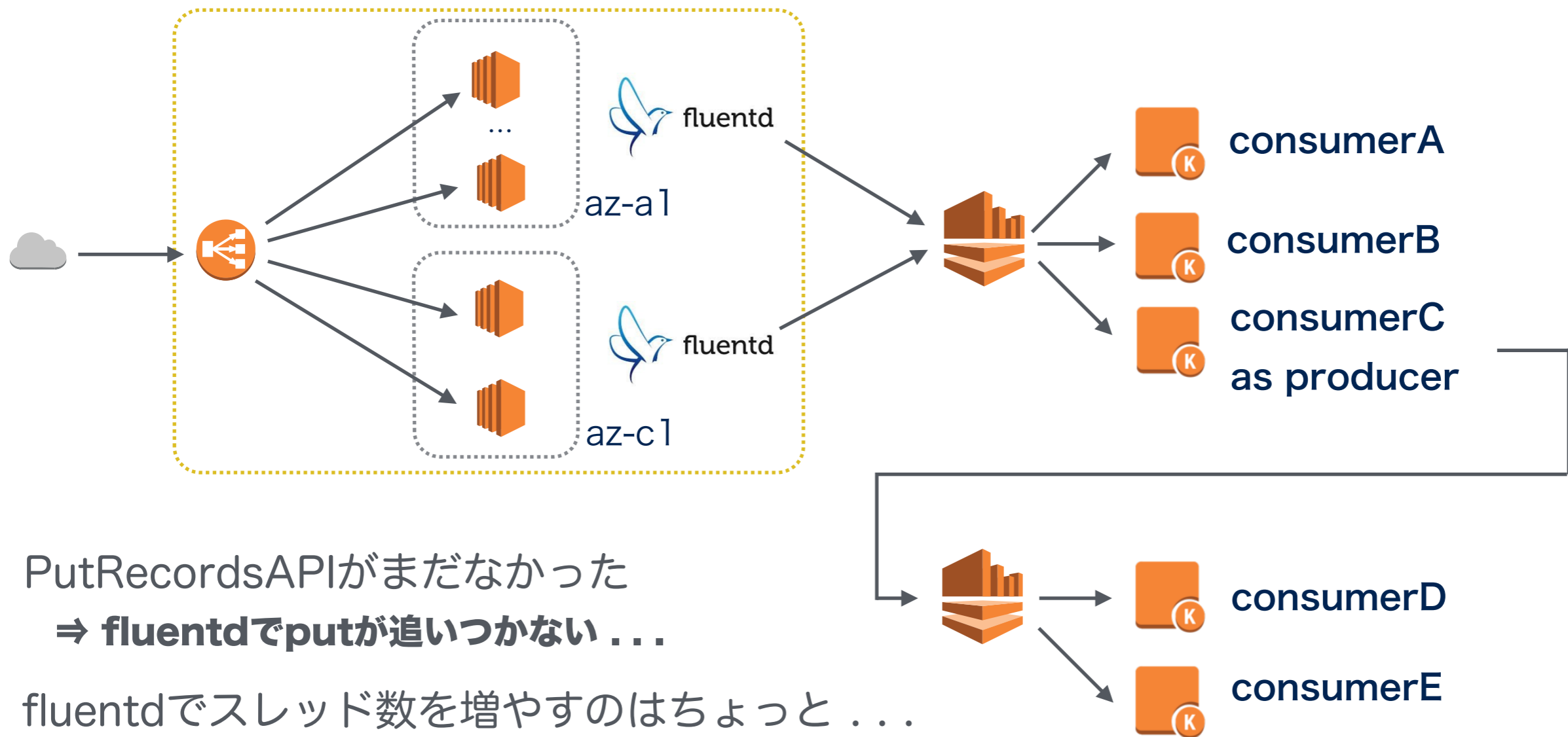
データ運搬フローの拡張 ~ 大まかな流れ ~



耐障害性の高いログ収集パイプラインとしての役割！

ログ収集サーバいらず、fluentdはbufferしてPutRecords

データ運搬フローの拡張 ~ 試行錯誤の過程 ~



PutRecordsAPIがまだなかった

⇒ **fluentdでputが追いつかない...**

fluentdでスレッド数を増やすのはちょっと...

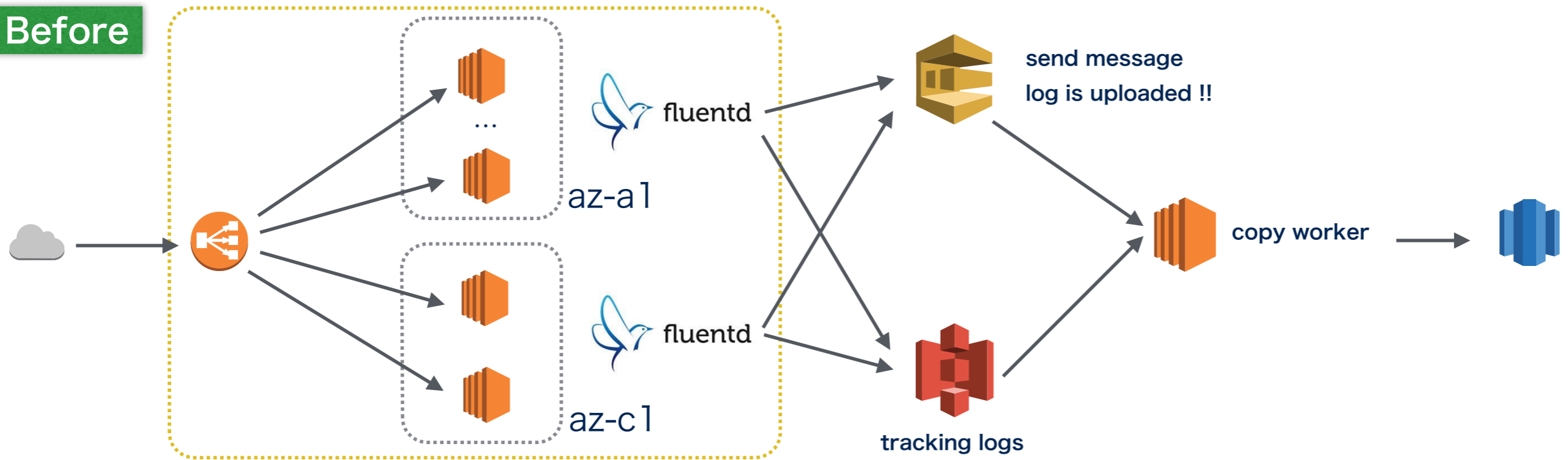
⇒ **アプリケーションサーバ内でのPutRecordで一時対処**

PutRecordsAPIがリリースされる ※ [AWS SDK for Ruby v2](#) から

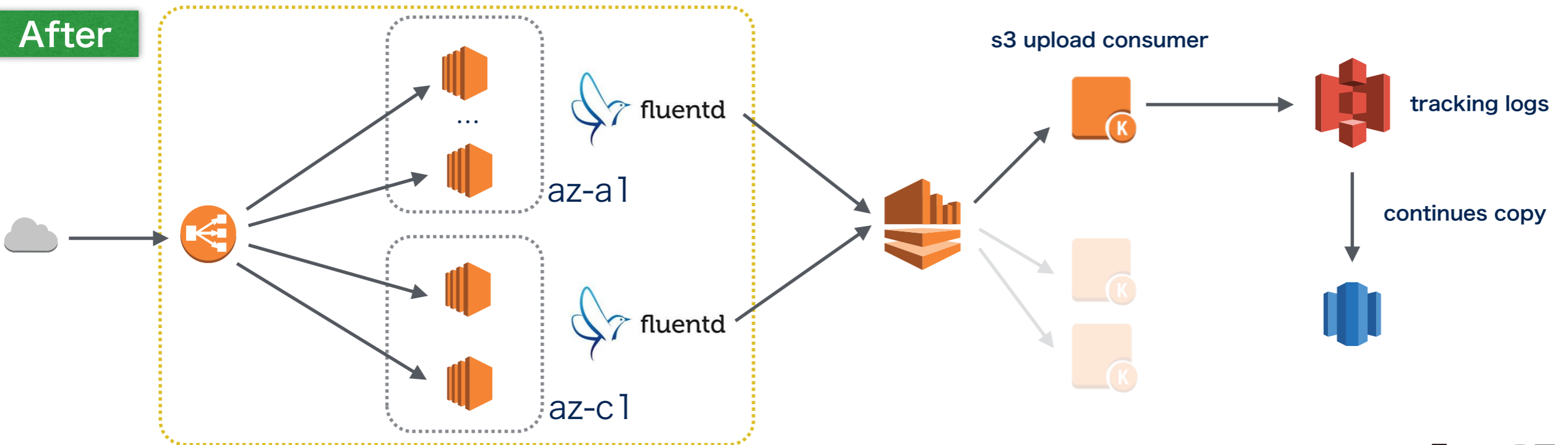
⇒ **どう考えてもbufferしてfluentdで処理させるのが安定する！**

データ運搬フローの拡張 ~ Before & After ~

Before



After



データ運搬フローの拡張 ~ Before & After ~

Before

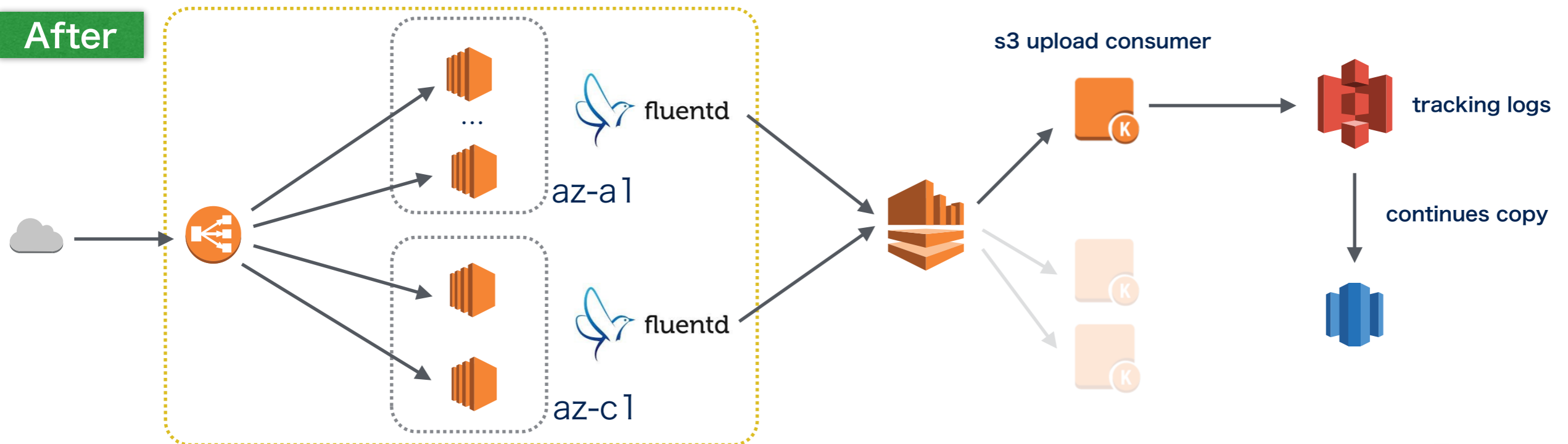
S3へのデータの運び方をKinesisを経由する方式に変更

⇒ Kinesis Consumerアプリを実装し、リアルタイムな何かをする環境を実現

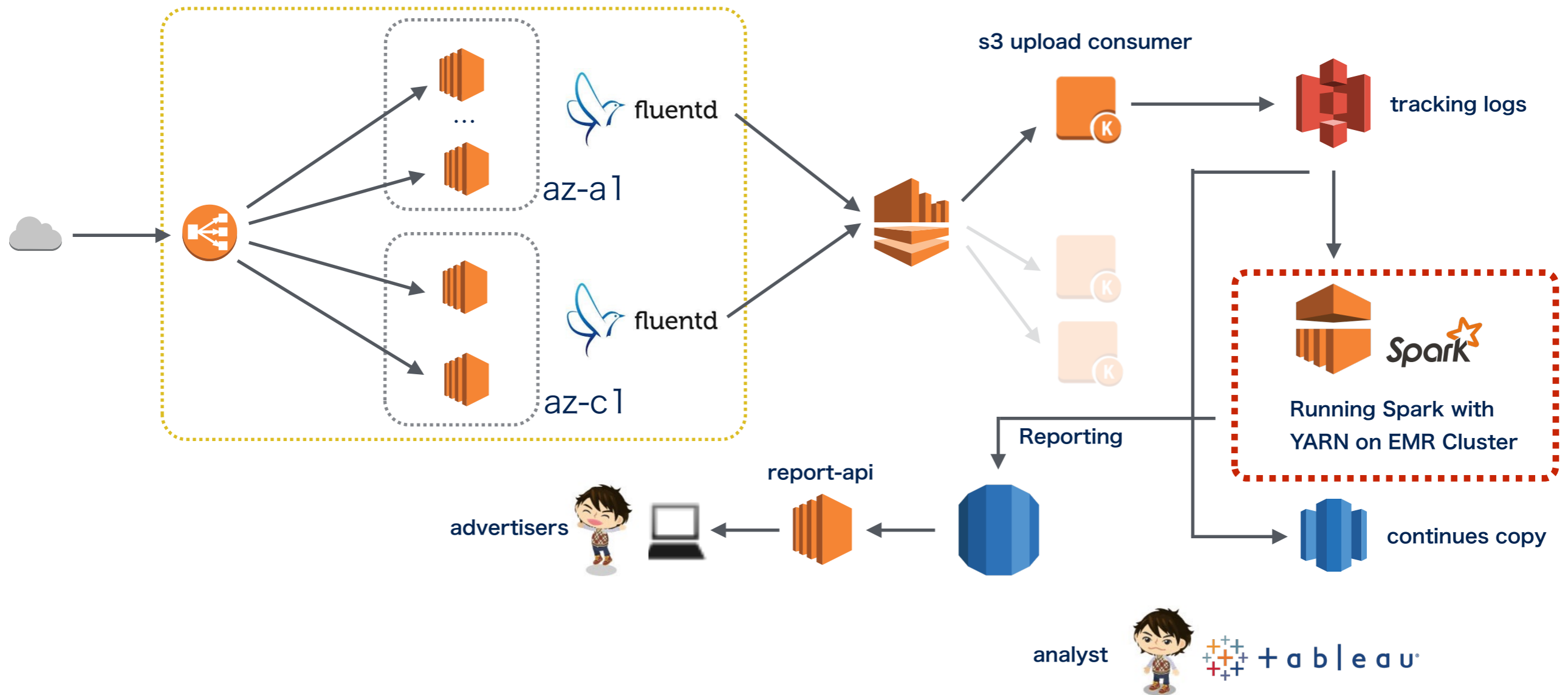
⇒ fluentdの役割の整理

- SQS, S3とのintegrationの解消
- Kinesis Producerとしての役割

After



分析基盤と集計基盤の役割の分離



定形レポート業務系処理をRedshiftからEMRへ

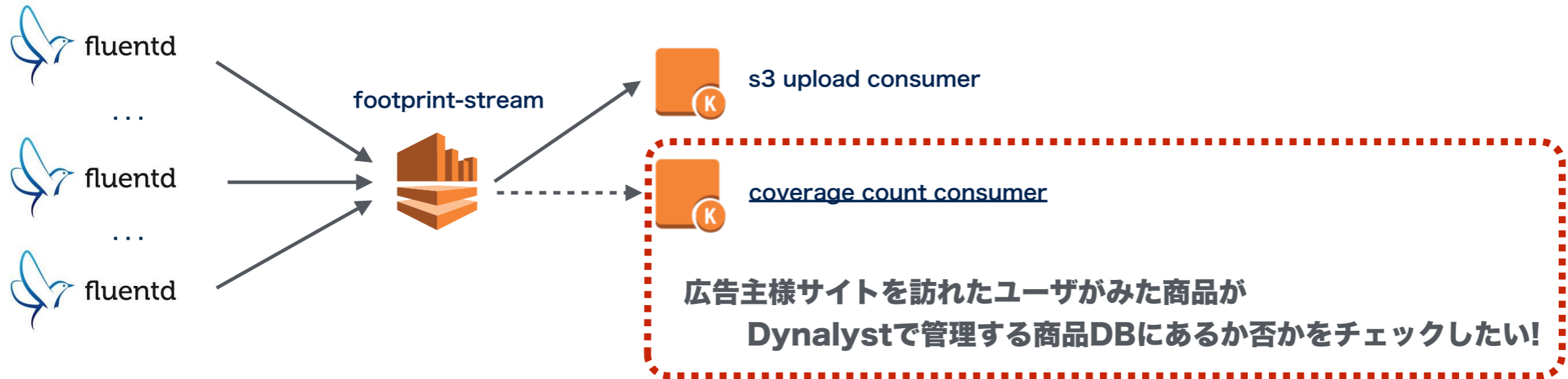
⇒ 分析系クエリを分析者がストレスなく投げられる環境の実現

⇒ Redshiftには定期的にCOPYしてログデータをためる

※ 知りたいことをSQL一本で記述できる点でRedshiftは分析者フレンドリー

※ 分析アプローチの傾向から意外と生ログを起点とした分析が多め

リアルタイムデータ処理アプリケーションの導入例



input:

- fluentd で PutRecords
- 秒間800 ~ 1000 recordくらいput
- shard数は2でstreamを作成

output:


- 2プロセスでconsumerをEC2上で実行させる
- scala app + KCLで実装
- daemon化してprocess管理
- streamから一括でGetRecordsし、
まとめて処理して結果をElastiCacheへ

リアルタイムデータ処理アプリケーションの導入例



KCLの挙動理解

- buildしたアプリのEC2へのデプロイ **proc1**
- 初回起動時にDynamoDBへcheckpoint管理用のtableが作成される
- チェックポイントを更新しながらアプリはshard2個分のデータを処理
- もう一つアプリをEC2にデプロイ **proc2**
- checkpoint管理テーブルを参照して、1個shardを奪って処理開始
- 1 shard-1 procで処理される状態になる
- proc2をshutdownするとproc1がshard2個分処理し始める



check point manage table

leaseKey	checkpoint	leaseOwner
shard-0000	seq-abcd123	consumer01:xxxx
shard-0001	seq-efgh456	consumer02:yyyy

※ **どのshardをどこまでだれ**が処理した？

※ **KCLは非常に高機能で便利**

リアルタイムデータ処理アプリケーションの導入例



KCL利用時のハマったポイント

- ピーク時に一方のprocが頑張り、他方のprocが頑張らない問題

=> DynamoDBのcheckpoint管理テーブルのWriteキャパオーバー

=> 初期がWriteCapacity10で作成されるので処理の単位、内容によっては10で足りないので増やす

- 新しいバージョンのアプリをデプロイしたらどうなるの？

=> proc3, proc4を起動, shard数2個の場合は古いproc1, proc2が仕事を奪われる

=> 実質rolling update的な挙動を示す

=> proc1, proc2を閉じてからproc3, proc4を起動したほうがいいときもあるかも？

- shard数が足りないのでapi経由でsplit-shardした、アプリはどうなる？

=> checkpoint管理テーブルがよしなにやってくれる

=> split-shardする前のshardの処理を終えたら新しいshardでの処理を引き継ぐ

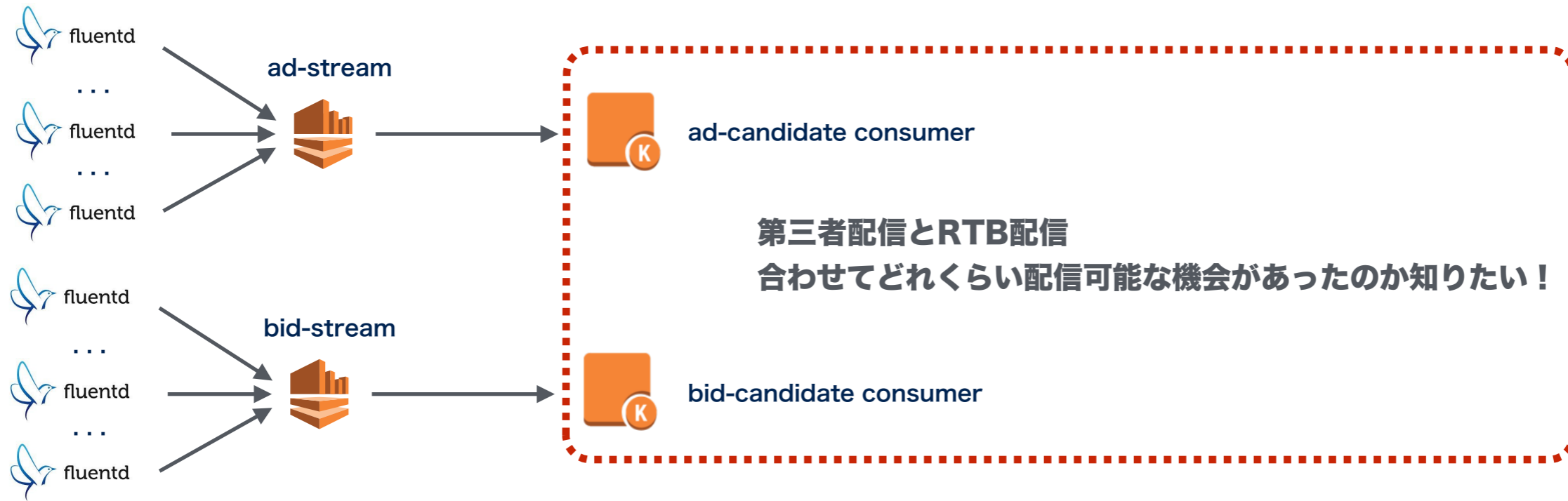
check point manage table

leaseKey	checkpoint	leaseOwner	parentShardId
shard-0002	seq-abcd123	consumer01:xxxx	["shard-0000"]
shard-0003	seq-efgh456	consumer02:yyyy	["shard-0000"]

※ このshardの親はだれ？

※ KCLは非常に高機能で便利

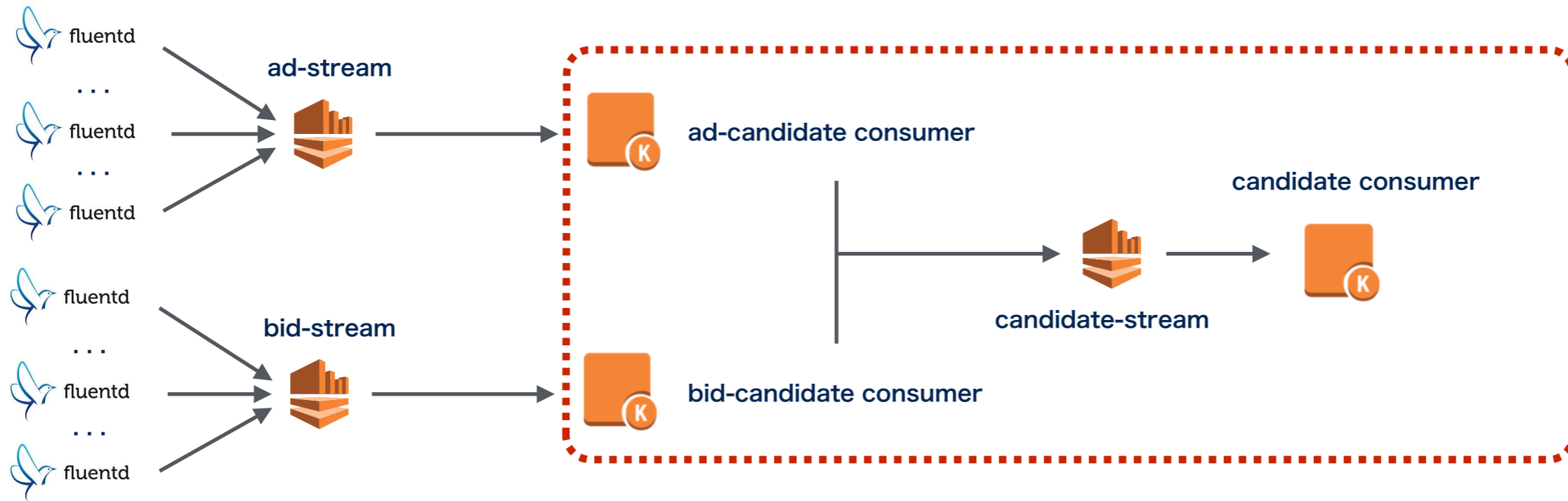
リアルタイムデータ処理アプリケーションの導入例



複数のstreamのデータを使いたい状況でどうする？

- ad-streamの流量は 2,000rps, shard数4
- bid-streamの流量は 40,000rps, shard数50

リアルタイムデータ処理アプリケーションの導入例



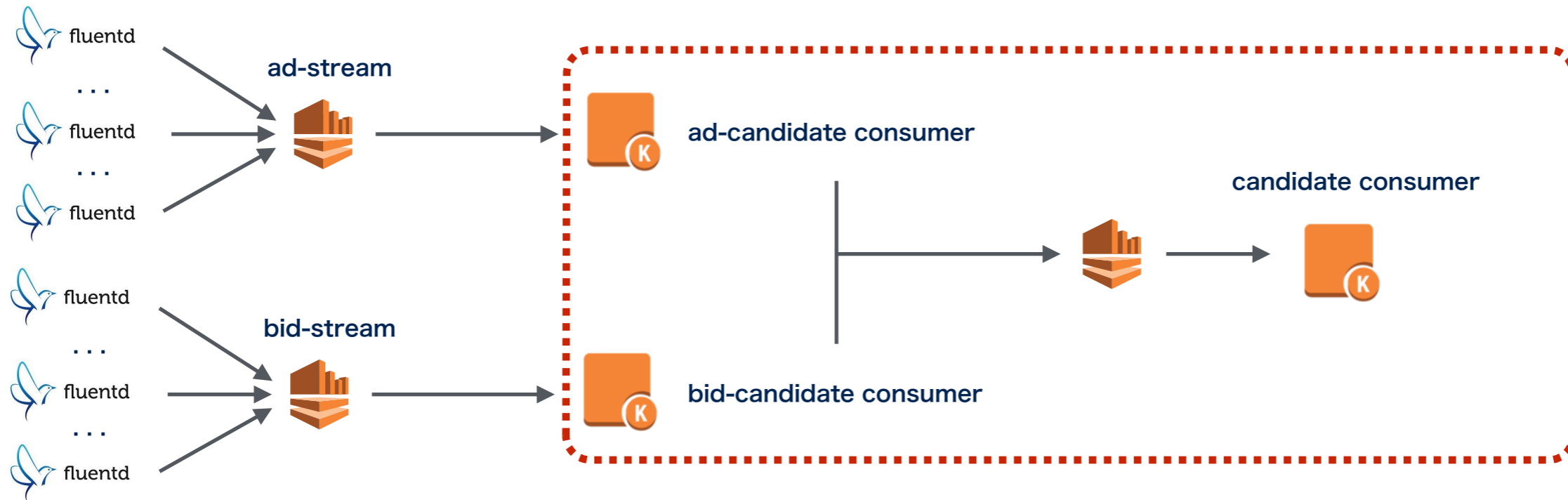
複数のstreamのデータを使いたい状況でどうする？

- ad-streamの流量は 2,000rps, shard数4
- bid-streamの流量は 40,000rps, shard数50

Dynalystでは、

- 1 consumerはstream毎にそれぞれ実装する**
- 2 consumerの中で別のstreamに再produceする処理を入れる**
- 3 candidate-streamのconsumerで目的の処理を実装する！**

リアルタイムデータ処理アプリケーションの導入例

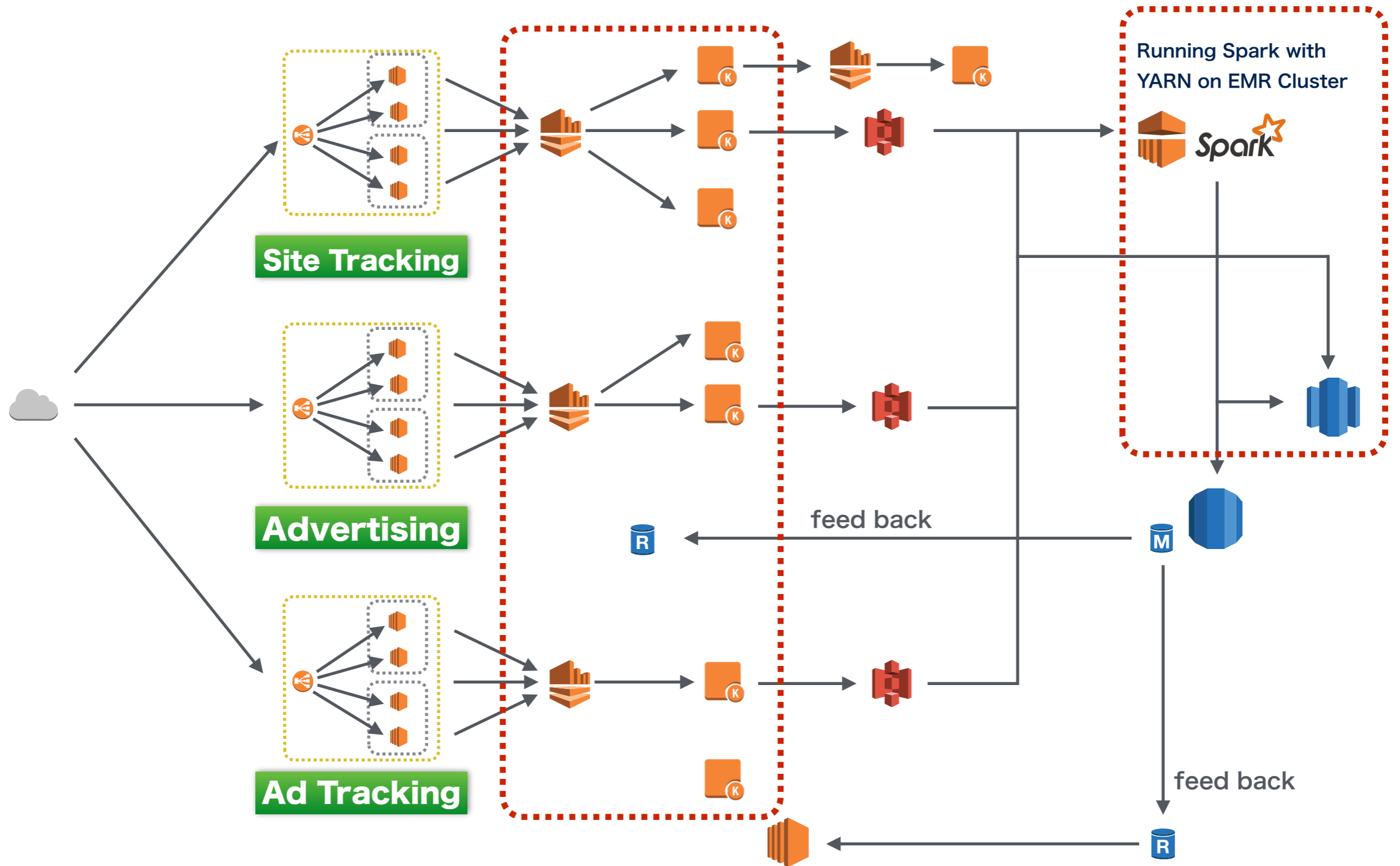


複数のstreamのデータを使いたい状況でどうする？

- ad-streamの流量は 2,000rps, shard数4
- bid-streamの流量は 40,000rps, shard数50

**GetRecordsしたものを集約し、
レコード数を減らして再produce !**

リアルタイムな処理とバッチ処理 ~ Overview ~



campaign management x reporting for Advertiser

Summary of this session

オンライン広告のいま

- 昨今の広告配信に求められることは何か？
- Dynalystが提供する価値とは何か？
- Dynalystのシステム構築におけるコンセプト

システムの大規模化に伴って直面した課題たち

- ログの量だけでなくログの内容が多様化することに対応しにくい問題
- データの利用用途の多様化と柔軟性の問題
- レポート集計基盤 ≠ アドホック分析基盤

Amazon Kinesisの採用とその導入の過程

- なぜKinesisで解決できると考えたか？
- Kinesisをコアとするデータ活用戦略と実現したシステムについて

ご清聴ありがとうございました