

トレーサビリティ (02A-OA)

プロジェクト	SE J 次世代システム構築プロジェクト
システム	次世代店舗システム

- 1_要件
- 2_方式概要
- 3_方式詳細
 - 3-1_店舗端末からAPIを実行するケース
 - 3-2_APIから非同期処理が実行されるケース
 - 3-3_APIから同一マイクロサービス内のAPIを実行するケース
 - 3-4_APIから店舗クラウド内の別マイクロサービスのAPIを実行するケース
 - 3-5_バッチからAPIを実行するケース
 - 3-6_ファイル格納を契機にAPIを実行するケース
 - 3-7_分散トレーシングログの出力内容

1_要件

- 特定のリクエストから店舗クラウド内における一連の処理を追跡できること。

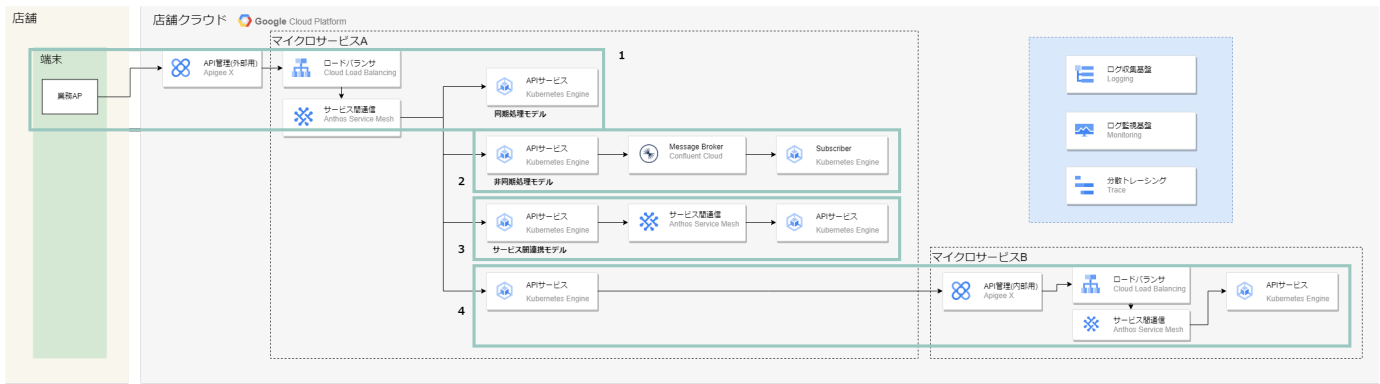
2_方式概要

- トレース情報(Trace ID、Span ID、Parent Span ID)をログ出力することにより、一連の処理を追跡する。
 - ログ出力時には、Span IDの親子関係を明示し(Span IDとParent Span ID)、サービス単位で処理が実行された順序が分かるようにする。
- API/SubscriberサービスにSpring Cloud Sleuthを組み込み、トレース情報を伝搬させる。

Trace ID	特定のリクエストを識別する一意なID。一連の処理を紐づけるために使用する。
Span ID	処理が実行されたAPI/Subscriberサービスを識別するID。一連の処理で一意な値とする。
Parent Span ID	ひとつ前のサービスで発行されたSpan ID。

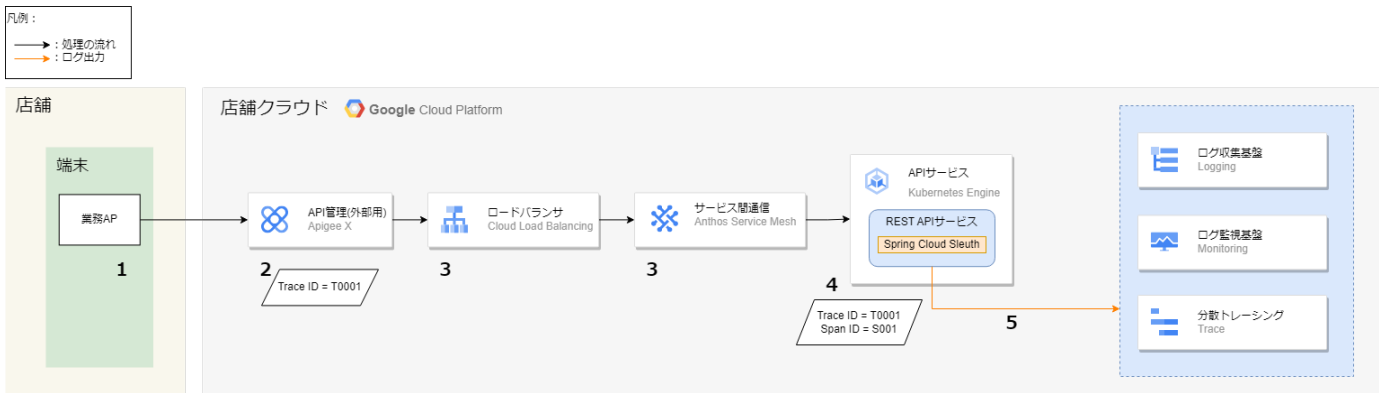
- モデルケースごとのトレース情報伝搬方式

No	モデルケース	伝搬方式概要
1	店舗端末からAPIを実行するケース	ApigeeXでTrace IDを発行し、APIサービスにHTTPヘッダーを介して伝搬する。
2	APIから非同期処理が実行されるケース	Spring Cloud SleuthのBaggageにトレース情報を設定して伝搬する。
3	APIから同一マイクロサービス内のAPIを実行するケース	HTTPヘッダーを介して、トレース情報を伝搬する。
4	APIから店舗クラウド内の別マイクロサービスのAPIを実行するケース	HTTPヘッダーを介して、トレース情報を伝搬する。
5	バッチからAPIを実行するケース	Step3で精査する内容。
6	ファイル格納を契機にAPIを実行するケース	Step3で精査する内容。



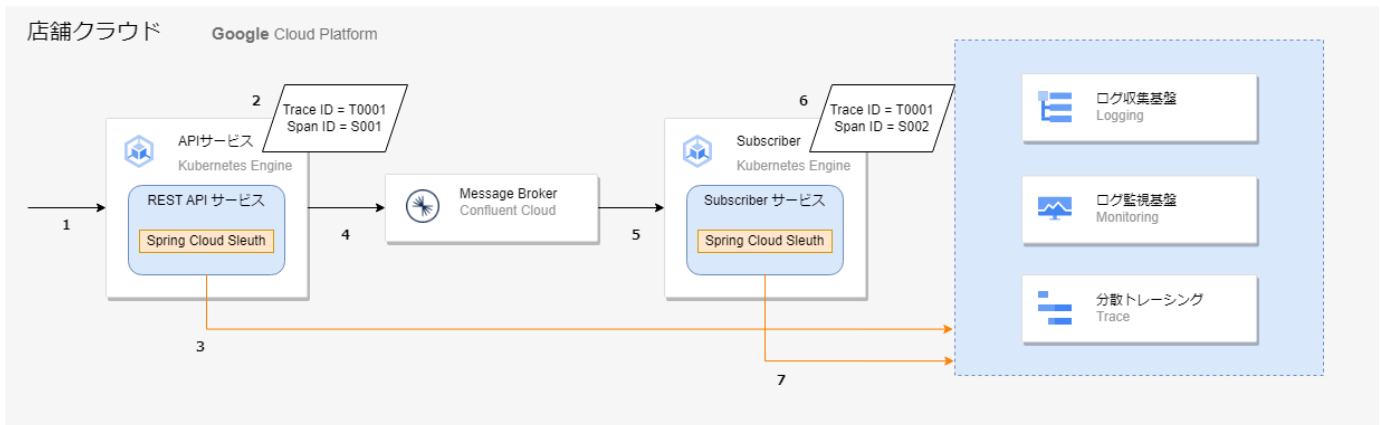
3_方式詳細

3-1_店舗端末からAPIを実行するケース



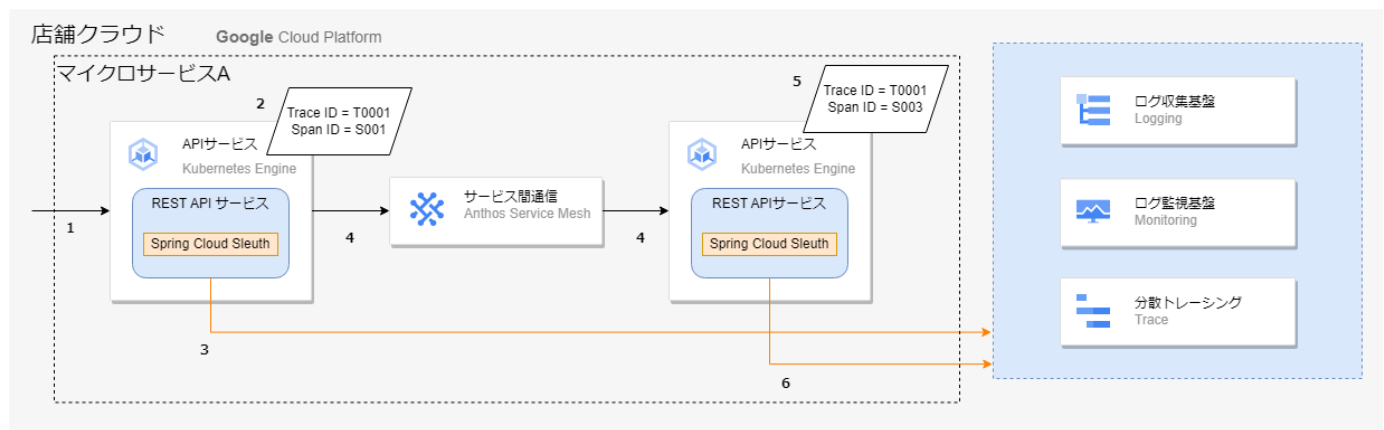
No.	説明
1	店舗端末からリクエストを送信する。
2	Apigee XでTrace IDを発行し、HTTPヘッダーを介してTrace IDを伝搬する。
3	HTTPヘッダーを介し、ロードバランサ、サービスメッシュ間で、Trace IDを伝搬する。
4	REST APIサービスでリクエストを受信する。Spring Cloud Sleuthで上流からのTrace IDを引き継ぎ、Span IDを新規発行する。
5	トレース情報をログ出力する。

3-2_APIから非同期処理が実行されるケース



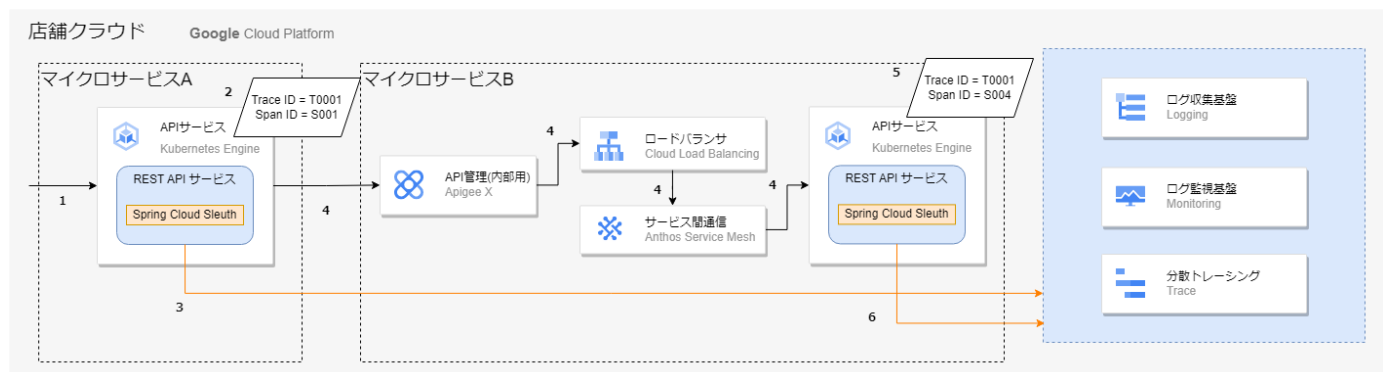
No.	説明
1	HTTPヘッダーにTrace IDが指定されたリクエストをREST APIサービスで受信する。
2	Spring Cloud Sleuthがトレース情報を発行する。Trace IDは上流から引き継ぎ、Span IDを新規発行する。
3	REST APIサービスからトレース情報をログ出力する。
4	Spring Cloud SleuthのBaggageにトレース情報を設定し、Confluent Cloudにメッセージをパブリッシュする。
5	Subscriberサービスがメッセージを受信し、Spring Cloud SleuthがBaggageからトレース情報を取り出す。
6	Spring Cloud Sleuthがトレース情報を発行する。Trace IDは上流から引き継ぎ、Span IDを新規発行する。
7	Subscriberサービスからトレース情報をログ出力する。

3-3_APIから同一マイクロサービス内のAPIを実行するケース



No.	説明
1	HTTPヘッダーにTrace IDが指定されたリクエストをREST APIサービスで受信する。
2	Spring Cloud Sleuthがトレース情報を発行する。Trace IDは上流から引き継ぎ、Span IDを新規発行する。
3	REST APIサービスからトレース情報をログ出力する。
4	トレース情報をHTTPヘッダーに設定し、サービスメッシュを介して、同一マイクロサービス内のREST APIサービスにリクエストを送信する。
5	Spring Cloud Sleuthがトレース情報を発行する。Trace IDは上流から引き継ぎ、Span IDを新規発行する。
6	REST APIサービスからトレース情報をログ出力する。


3-4_APIから店舗クラウド内の別マイクロサービスのAPIを実行するケース




No.	説明
-----	----

1	HTTPヘッダーにTrace IDが指定されたリクエストをREST APIサービスで受信する。
2	Spring Cloud Sleuthがトレース情報を発行する。Trace IDは上流から引き継ぎ、Span IDを新規発行する。
3	REST APIサービスからトレース情報をログ出力する。
4	トレース情報をHTTPヘッダーに設定し、ApigeeX(内部用)、ロードバランサ、サービスメッシュを介して、別マイクロサービスのREST APIサービスにリクエストを送信する。
5	Spring Cloud Sleuthがトレース情報を発行する。Trace IDは上流から引き継ぎ、Span IDは新規発行する。
6	別マイクロサービスのREST APIサービスからトレース情報をログ出力する。

3-5_バッチからAPIを実行するケース

 Step3で検討する。

3-6_ファイル格納を契機にAPIを実行するケース

 Step3で検討する。

3-7_分散トレーシングログの出力内容

- Cloud Traceにおけるログ出力内容

項目	出力内容	出力例
ログレベル	ログの緊急度や重要度、用途が分かるように出力する	INFO
タイムスタンプ	処理の開始時刻、ログに書き込まれた時刻が分かるように出力する	Oct 12 19:27:09.613
処理時間	タイムスタンプより処理にかかった時間を割り出し出力する	3 ms
トレースID	どのリクエストに紐づくログなのかわかるように出力する	382d4f4c6b7bb2f4a972559d9085001d
スパンID	そのエンドポイントの処理IDを出力する	2205310701640571284
親スパンID	そのエンドポイントを直接呼び出した処理IDを出力する	3426592852948343287
サービス名	どのサービスに関連するログかわかるように出力する	xxxx API

版数	改版日	改版目的	改版箇所・改版内容	対応者
1.00	2023/01/04	新規作成	—	NESD 今泉