

バッチ処理でhakoを使う話

 @hoshinotsuyoshi / feedforce.Inc.

2016/10/12

Me

-  @hoshinotsuyoshi
-  
-  
- 2014～ feedforce.Inc.

バッチ処理 🕒

- cronだと
 - 高い可用性を得るのが難しい
 - リソースに応じたスケジューリングが難しい



ECS

EC2 Container Service Product Details

The background features a series of 3D rectangular blocks in shades of orange and grey, arranged in a stepped, architectural pattern. The blocks vary in height and are positioned to create a sense of depth and structure, with some blocks appearing to support others.

“ You can run anything: applications, **batch jobs**, or microservices. ”

ECSでバッチ処理するには

1. TaskDefinitionの登録して

- `$ aws ecs register-task-definition`

2. Taskを実行する！

- `$ aws ecs run-task`

3. 以上！(基本的には)

その他の問題

- オートスケーリングさせなきゃ... 🥲
- TaskDefinitionが際限なく増加してく... 🥲
- デプロイのたびにTaskDefinition登録してた
- 登録済みかどうかのチェックがだるい
- ...

2016年9月

? 「hakoのoneshot  良さそう」



Search Gems...



GEMS

GUIDES

hako *0.21.0*

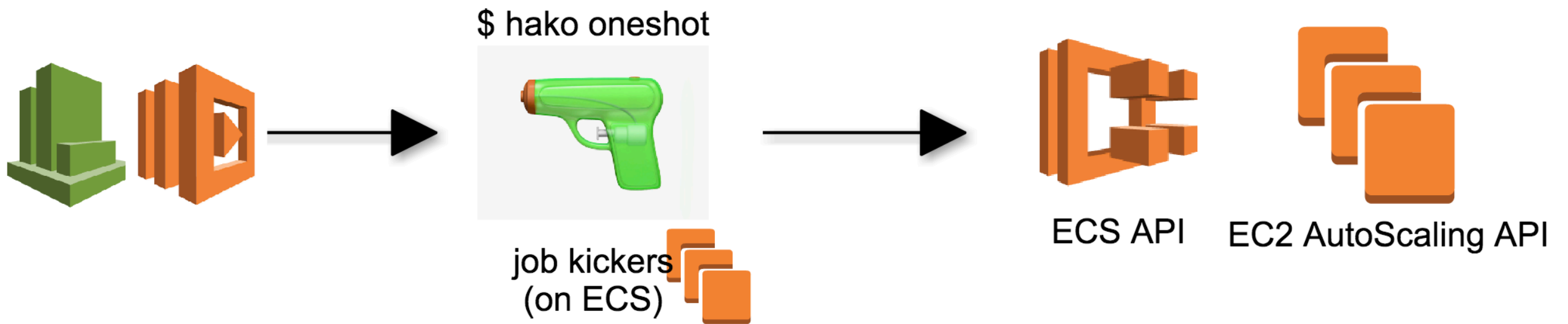
Deploy Docker container

\$ hako oneshot 🚬

“ hako oneshot はYAML の定義に従って ECS の RunTask API を呼び出すコマンド ”

- ECS を利用したオフラインジョブの実行環境 - クックパッド開発者ブログ

こんなかんじになった



🔫 の中身



job kickers
(on ECS)

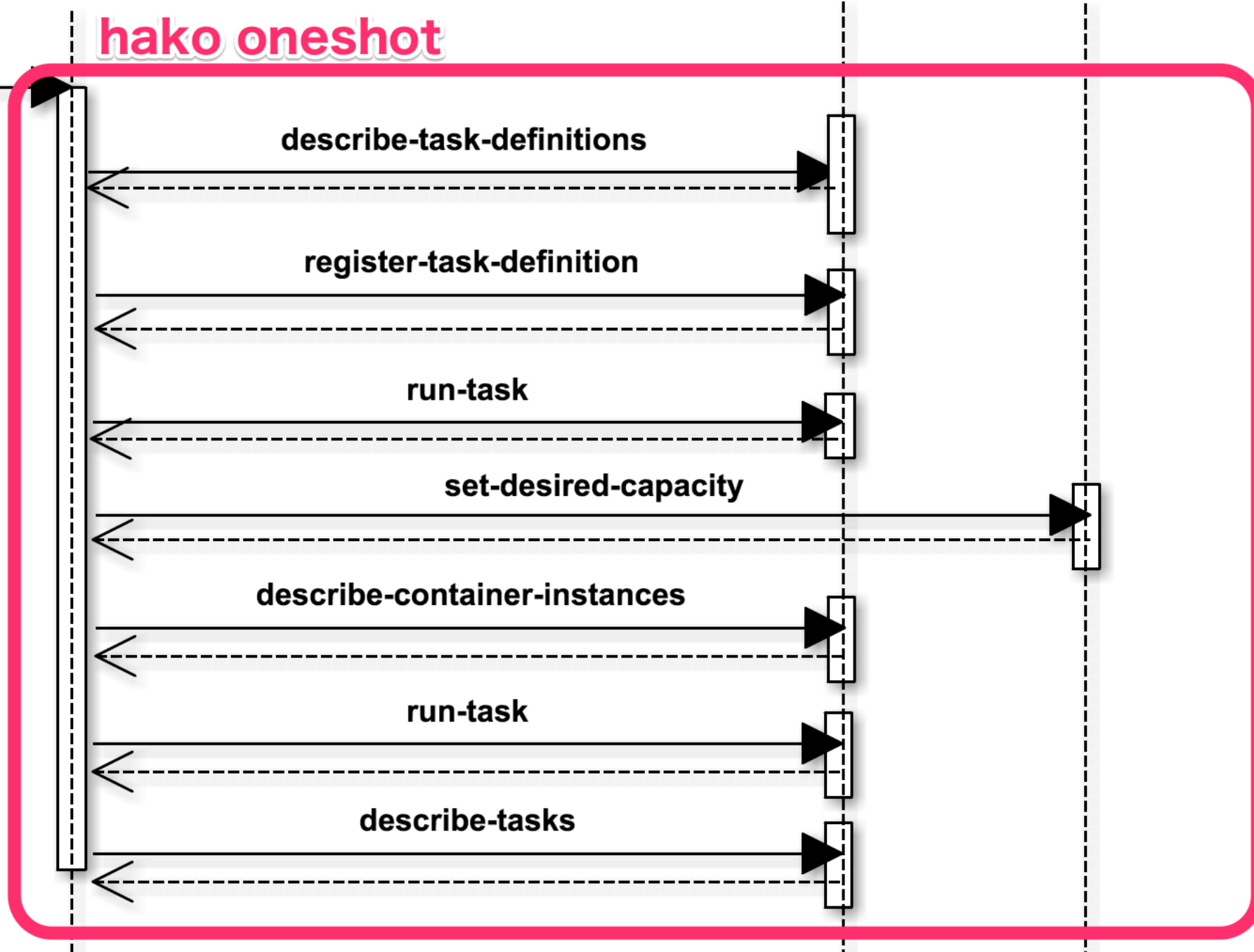
hako oneshot

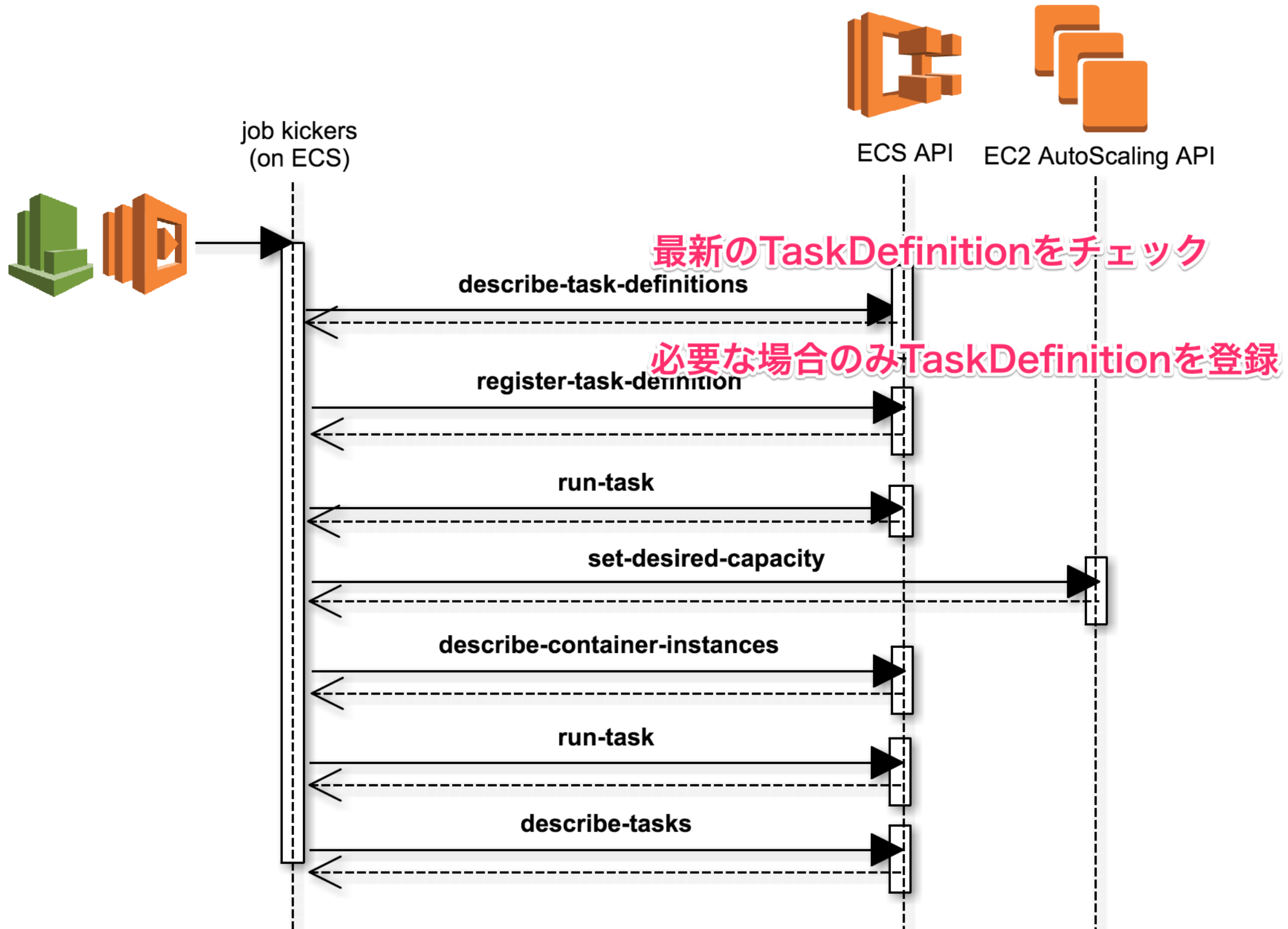


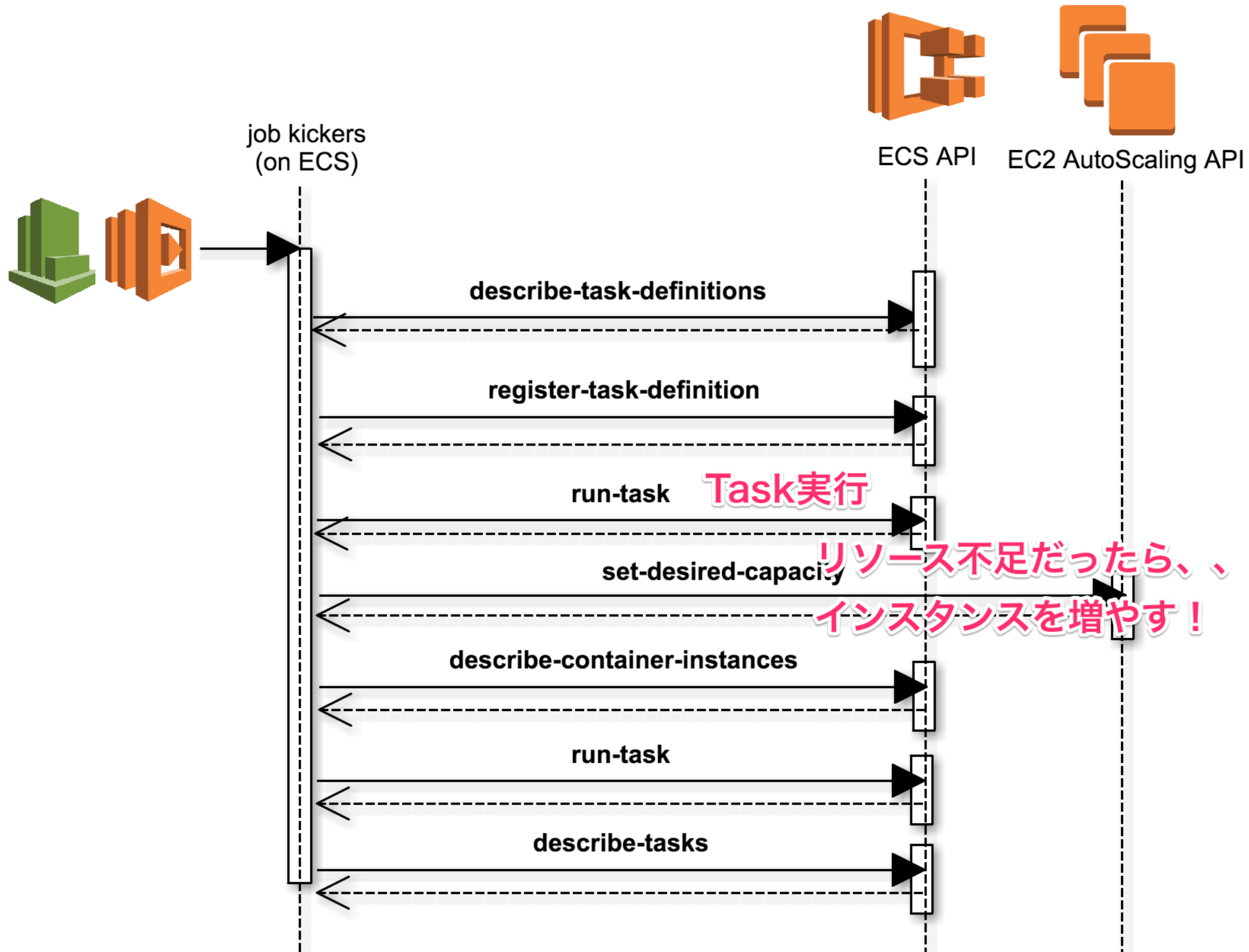
ECS API

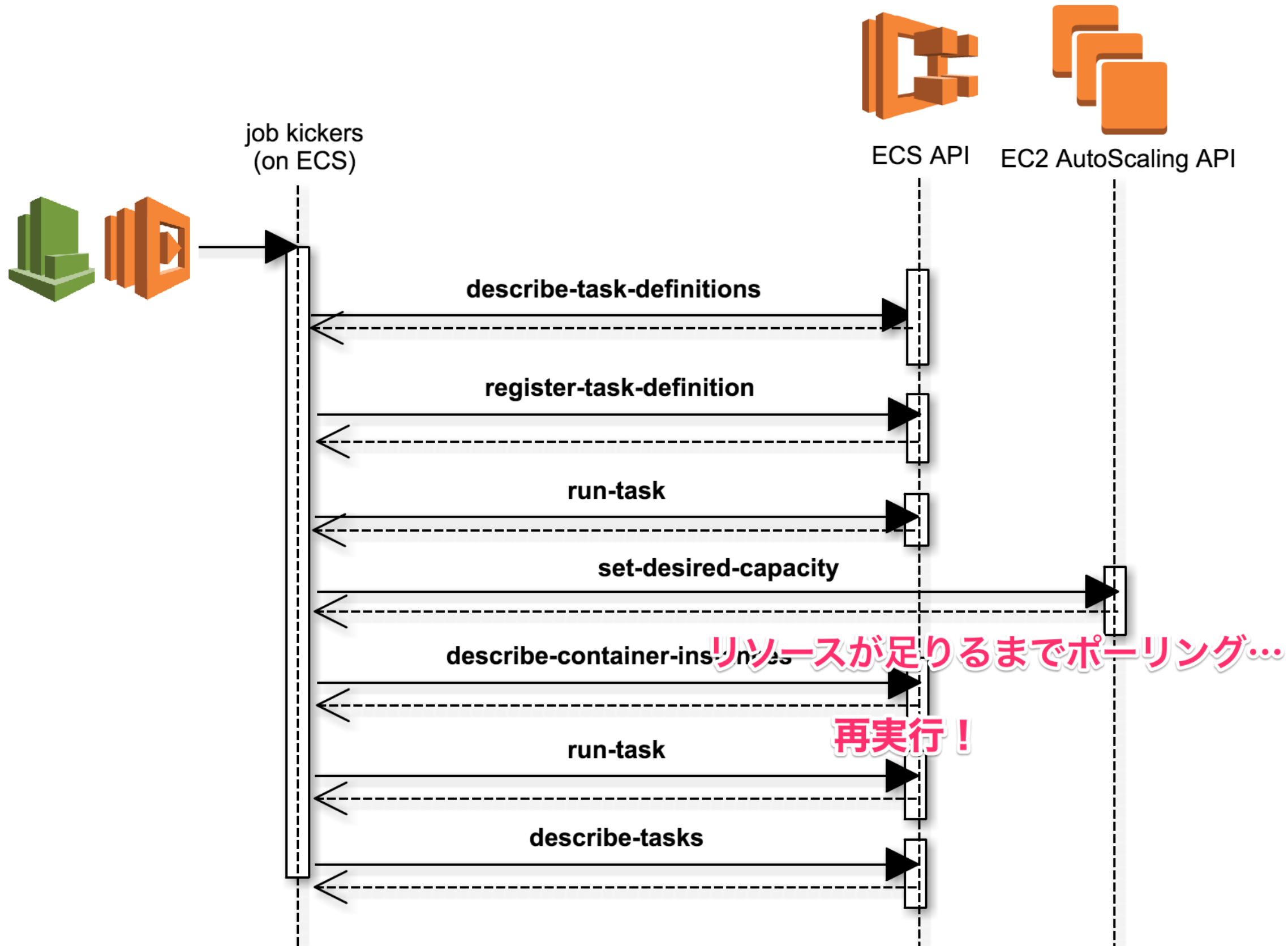


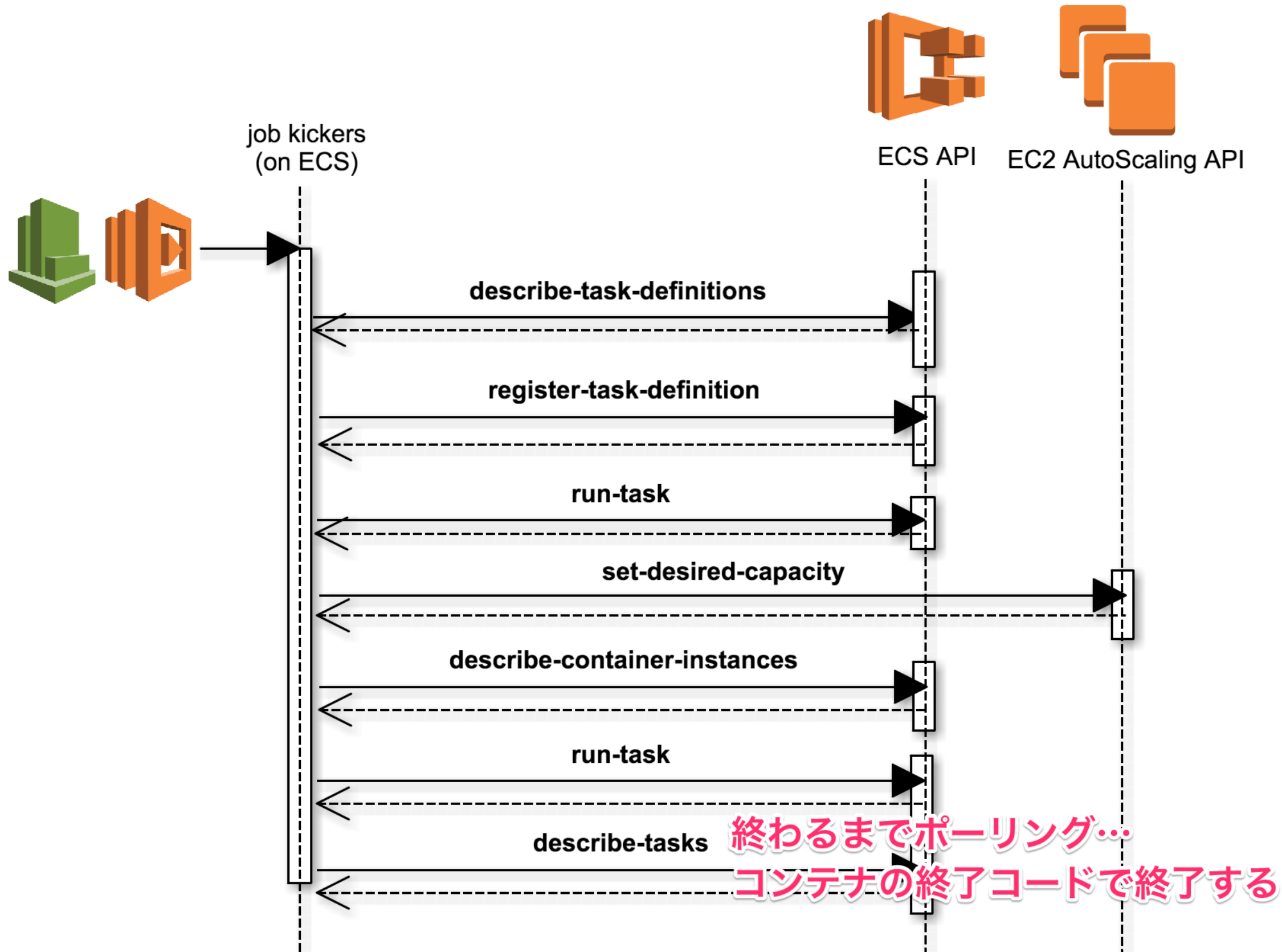
EC2 AutoScaling API











TaskDefinitionが増えづらい

- TaskDefinitionを重複登録しない仕組みがある
 - (たまに登録されちゃうパターンがある)
 - (ソース読んで直したり、yamlの書き方で回避する)

自動scale out

- リソースが足りないときの自動scale out
- `autoscaling_group_for_one-shot` を指定すればおk

`scheduler:`

`...`

- yaml例) `autoscaling_group_for_one-shot:` (オートスケーリンググループ名)

- リソースが足りない場合はdesiredが **1** 増える
- この **1** はハードコード

まとめ

- hako oneshot 🚬 を使うことにより
 - 余分なTaskDefinition登録が減った
 - 自動scale outも面倒見てくれる
 - ECS API叩く **自前コードが減った**

参考文献

- AWS _ Amazon EC2 Container Service _ Product Details
 - https://aws.amazon.com/ecs/details/?nc1=h_ls
- eagletmt/hako
 - <https://github.com/eagletmt/hako>

参考文献

- ECS を利用したオフラインジョブの実行環境 - クックパッド開発者ブログ
- <http://techlife.cookpad.com/entry/2016/09/09/235007>