**🚀 EC2 と Fargate の違い**

AWS ECS（Elastic Container Service）では、**「EC2モード」** と **「Fargateモード」** の2つの実行方法があります。

■EC2が向いているケース

・OSやネットワークの細かいチューニングが必要

・大規模でサーバー管理工数以上にAWSコスト削減のメリットが大きい

・GPUを使う必要がある

■Fargateが向いているケース

・OSやネットワークのチューニング要件がない

・サーバー管理をする人材がいない、予算をかけたくない

それぞれの **違い** をわかりやすく説明します！✨

**📌 1. EC2（Amazon Elastic Compute Cloud）とは？**

**🖥 自分でEC2インスタンス（仮想マシン）を管理し、その上でコンテナを実行する方法**

📌 **特徴**

* **EC2（仮想サーバー）を自分で起動・管理** する
* コンテナを実行するための **インフラ（OS・CPU・メモリ）を管理** する
* **自由にカスタマイズ可能**（インスタンスの種類・OS設定・ネットワーク設定）
* **コスト管理がしやすい**（EC2の料金体系に基づく）

📌 **使いどころ** ✅ **カスタマイズが必要な場合**（特殊なミドルウェア・カーネルチューニング）  
✅ **長時間稼働するアプリ**（EC2の料金モデルが有利）  
✅ **既存のオンプレミス環境と統合する場合**

📌 **EC2のイメージ**

* **EC2は「自前のサーバー」に近い**
* **OSやインスタンスサイズを自由に設定可能**

**📌 2. Fargate（AWS Fargate）とは？**

**☁️ AWSがインフラ（サーバー）を完全に管理し、コンテナだけを実行できる「サーバーレス」な方法**

📌 **特徴**

* **EC2を用意する必要なし！**（AWSがすべて管理）
* **サーバーレスなコンテナ実行環境**（OS・インスタンス管理不要）
* 必要なCPU・メモリだけを指定するだけでOK！
* **スケーラビリティが高い**（負荷に応じて自動調整）

📌 **使いどころ** ✅ **インフラ管理をしたくない場合**（EC2の管理が不要）  
✅ **スケールが必要なアプリ**（負荷に応じて増減）  
✅ **短時間実行のワークロード**（課金は実行時間のみ）

📌 **Fargateのイメージ**

* **Fargateは「レンタルキッチン」みたいなもの**
* 必要なコンテナリソースだけ指定し、**あとはAWSがすべて管理**

**📌 3. EC2 vs Fargate の違い（比較表）**

| **項目** | **EC2 モード** | **Fargate モード** |
| --- | --- | --- |
| **インフラ管理** | **必要（EC2の管理が必要）** | **不要（AWSが管理）** |
| **スケーリング** | 自分で設定（Auto Scaling） | 自動スケーリング |
| **OSの管理** | 必要（パッチ適用・更新） | 不要（AWSが管理） |
| **起動時間** | 数分かかる | 数秒で起動 |
| **コスト** | **固定費**（EC2のインスタンス費用） | **従量課金**（使用した時間だけ） |
| **柔軟性** | **高い（カスタマイズ可）** | **低い（AWSの仕様に依存）** |
| **ユースケース** | **長時間動かすアプリ**、カスタマイズが必要 | **短時間タスク**、サーバーレス |

**📌 4. どっちを選ぶべき？**

**✅ EC2モードを選ぶべきケース**

* **コンテナが長時間動く**（常に稼働するWebサービスなど）
* **特殊な環境が必要**（カーネル変更・ストレージ設定など）
* **固定費でコストを最適化したい**

**✅ Fargateモードを選ぶべきケース**

* **インフラ管理をしたくない**（EC2の運用をAWSに任せたい）
* **スケールが必要なアプリ**（トラフィックに応じて自動調整）
* **バッチ処理などの一時的なタスク**

**📌 5. 実際の活用例**

| **ユースケース** | **適した実行モード** | **理由** |
| --- | --- | --- |
| **常時稼働するWebアプリ** | **EC2** | 常に動かすならEC2のほうがコスト最適化しやすい |
| **負荷変動のあるアプリ** | **Fargate** | トラフィックに応じて自動でスケール |
| **バッチ処理・ETL** | **Fargate** | 処理が終われば課金が止まるため、コスト効率が良い |
| **機械学習の推論API** | **EC2** | GPU付きEC2インスタンスを選べる |
| **開発・テスト環境** | **Fargate** | 短時間だけ実行し、終わったら課金なし |

**📌 6. まとめ**

| **EC2 vs Fargate** | **どんなときに使う？** |
| --- | --- |
| **EC2モード** | **インフラを自由に管理したい・コストを抑えたい・長時間稼働** |
| **Fargateモード** | **インフラ管理を省略したい・スケールが必要・短時間のタスク** |

💡 **簡単に始めるなら「Fargate」！インフラを細かく管理したいなら「EC2」！**

どちらを選ぶかは **「管理の自由度」と「コスト最適化」** のバランス次第です！🚀