クラウド IDE とエミュレータを利用した実機レス開発環境

アイコムシステック株式会社 庭野 正義

2020/6/12

背景と目的

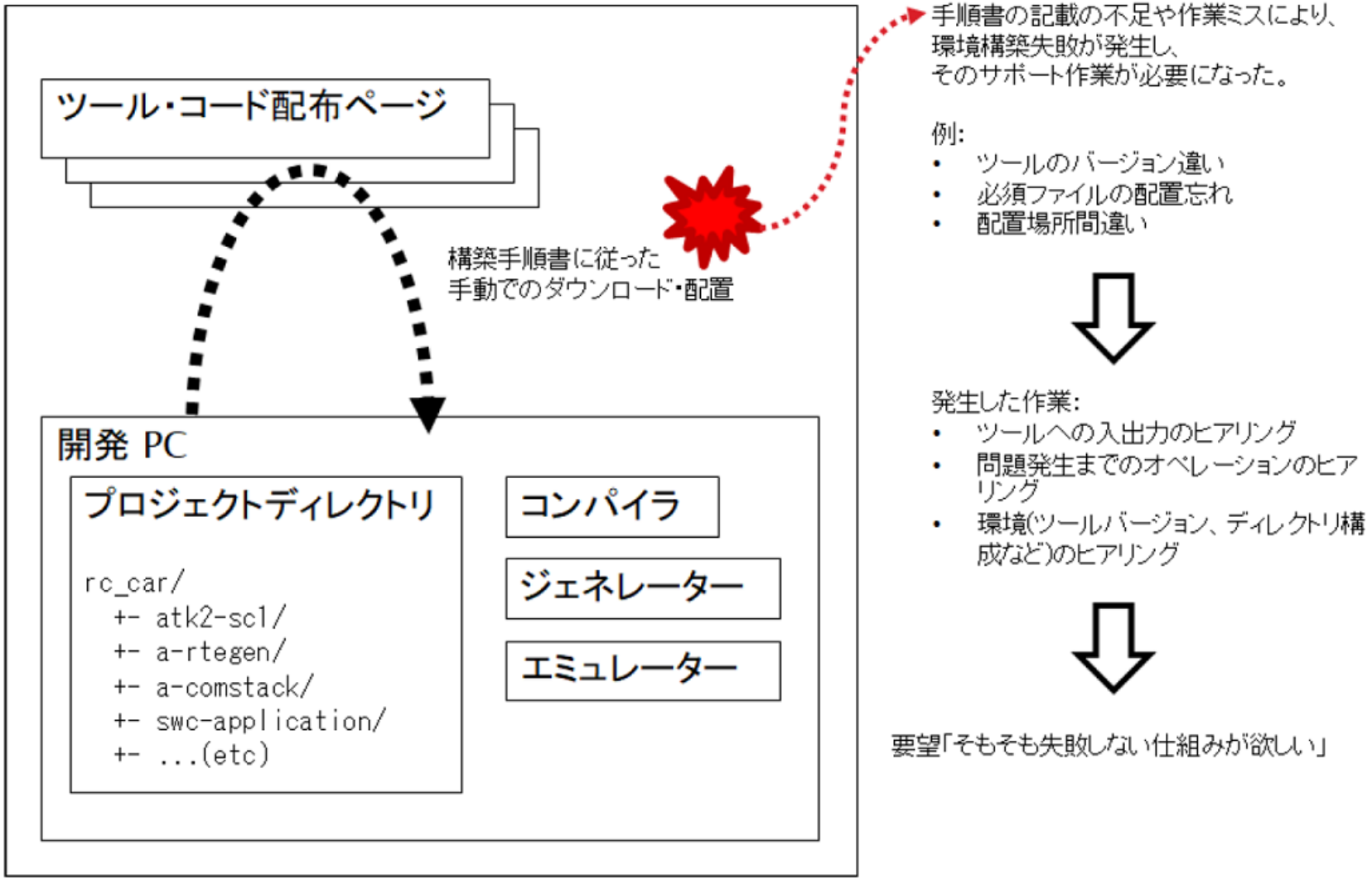


図1: 背景と目的

クラウド IDE 開発環境概要

今回提案したアイデアでは、クラウド IDE に Eclipse Che※1 を使用し検証した。

利用者が Eclipse Che 上でどのように操作するかを説明する。

※1: https://www.eclipse.org/che/

開発手順(I) -ワークスペーステンプレート選択-

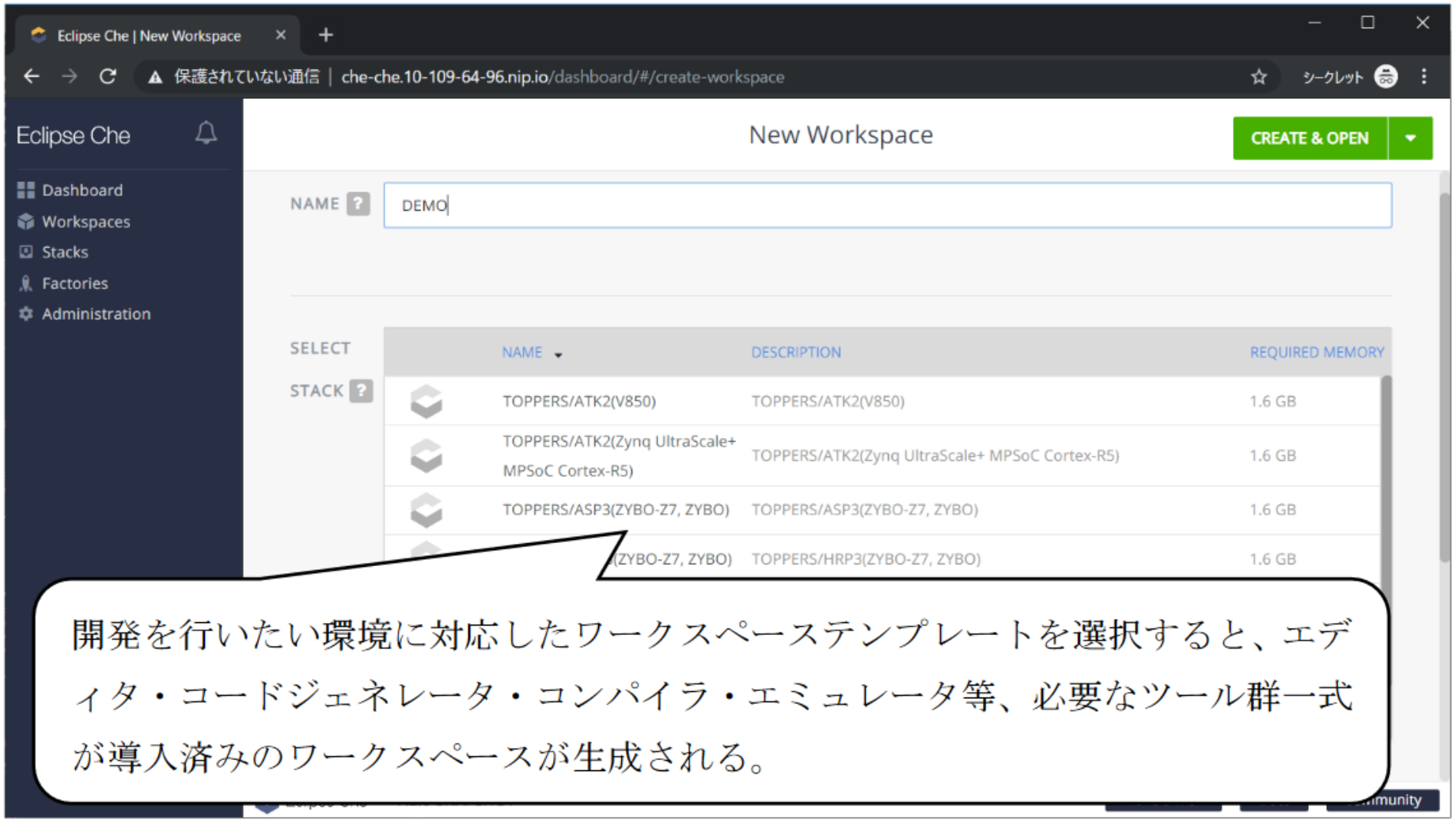


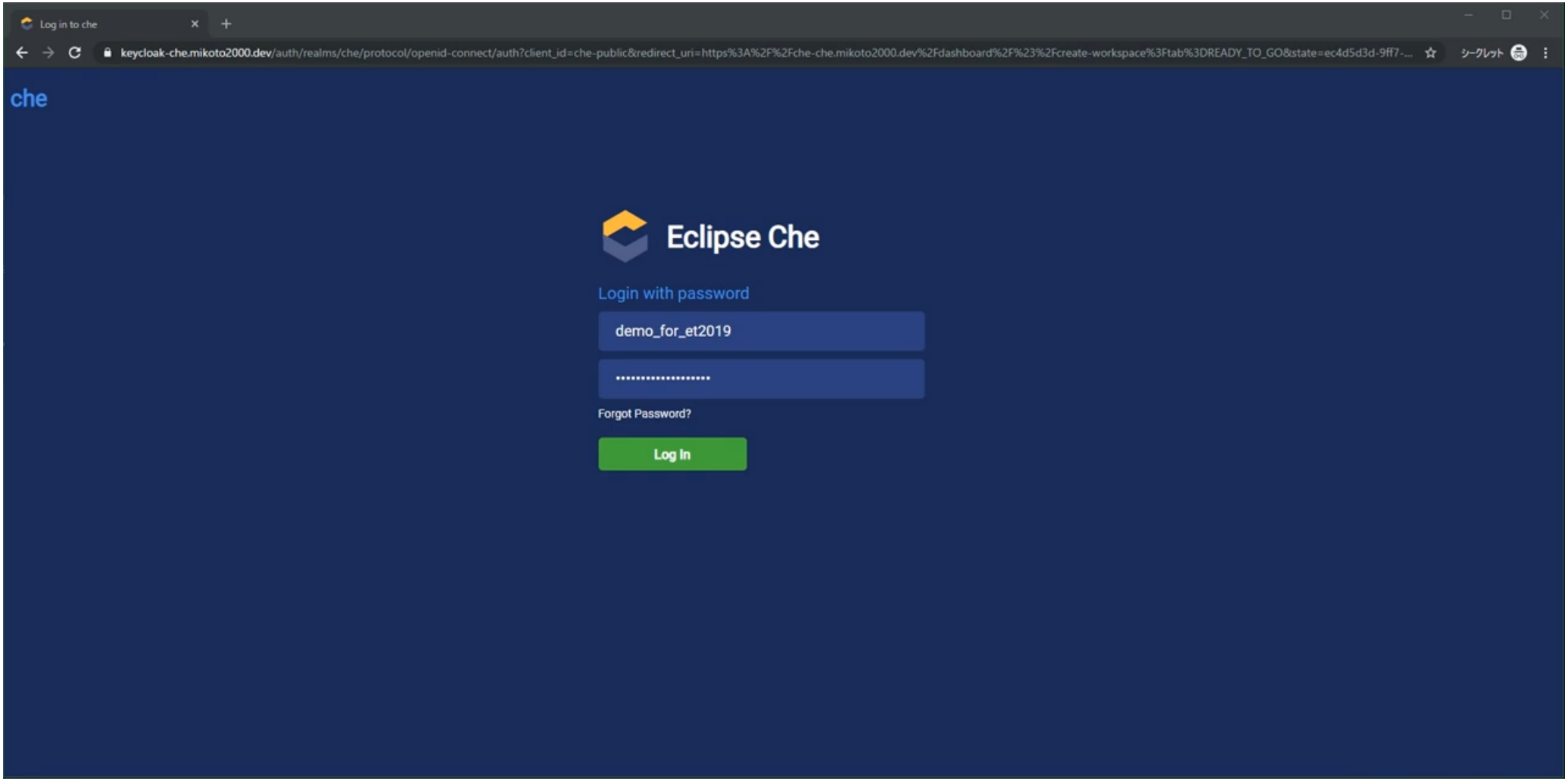
図2: クラウド IDE オーバービュー

開発手順(2) -プロジェクト編集-

図3: クラウド IDE オーバービュー

クラウド IDE で TOPPERS カーネルを動かす

DEMO.



構成 -概要-

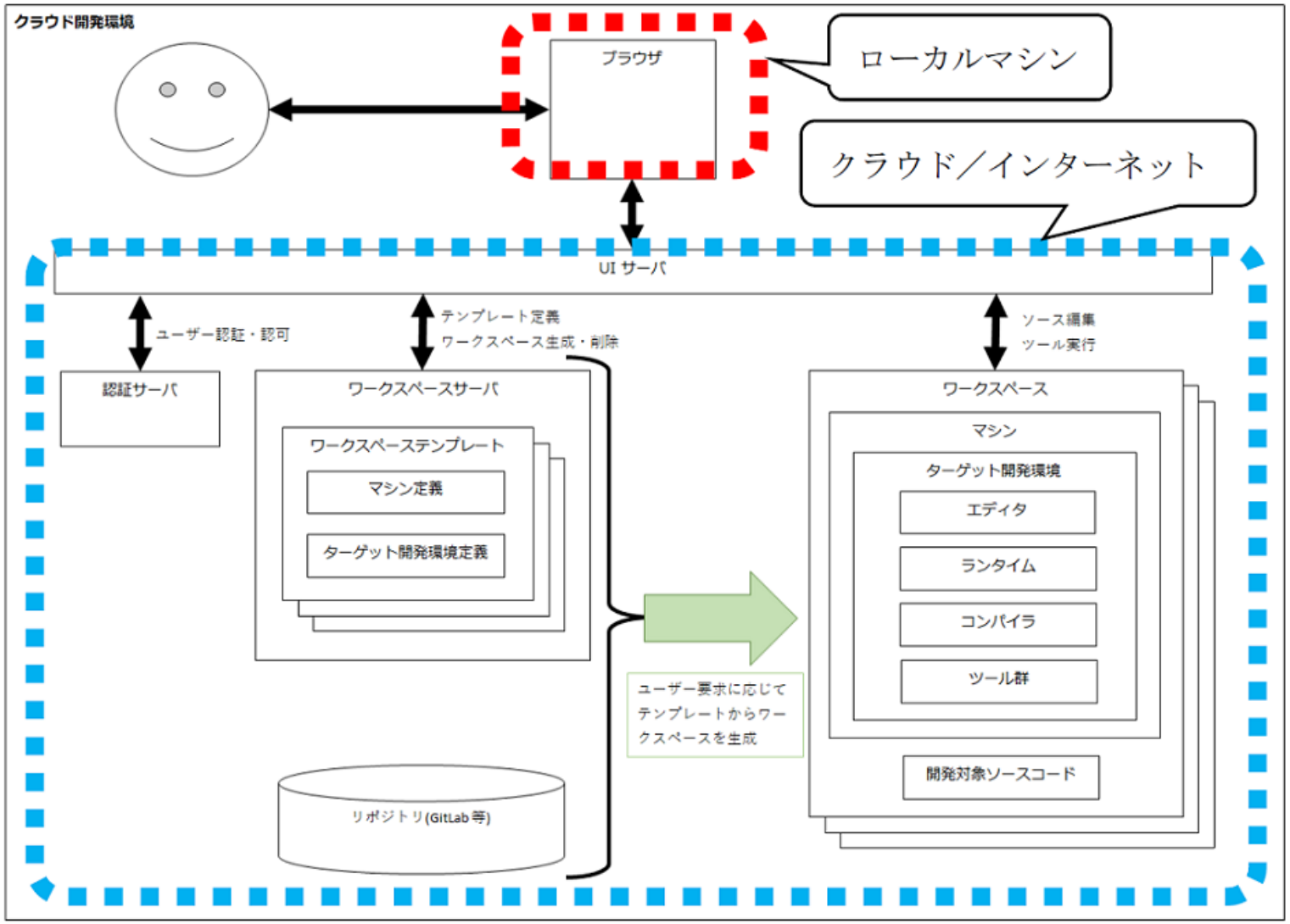


図4: クラウド IDE オーバービュー

独自開発環境構築例

DEMO.

mikoto2000/che-aarch64-simple-sample を見ながら、 che.openshift.io 上へ独自開発環境を展開し、動作確認を行う。

今後の展望

1. CI サービスとの連携

    - *コンテナベースの CI サーバーを使用すれば、 Eclipse Che 用に作成した Docker コンテナを流用し、それをそのまま使って CI を行うことができるはず*
    - *CI サーバーとの連携のノウハウ・ガイドラインを整備すれば、より効率よく CI を実施することができる。そうすれば、教育だけでなく、開発にも本格的に導入していくことができるかもしれない*

2. ワークスペーステンプレート集の作成

    - *「独自開発環境構築例」で示した通り、開発環境を定義した yaml を公開しておけば、Eclipse Che に読み込ませることができる*
    - *開発環境定義の yaml を集めた GitHub リポジトリを作成すれば、より簡単に試してもらうことができるようになると思われる*

参考資料

- 基本情報
  - *Eclipse Che | Eclipse Next-Generation IDE for developer teams*
  - *Introduction to Eclipse Che | Eclipse Che Documentation*
- インストール
  - *Running Che locally | Eclipse Che Documentation*
  - *Running Eclipse Che on Kubernetes using Docker Desktop on macOS or Windows*
  - *Windows の Docker Desktop(with Kubernetes) に Eclipse Che をデプロイした - mikoto2000 の日記*
- 独自開発環境の追加
  - *Overview | Eclipse Che Documentation*
  - *Eclipse Che に自作の Che Plugin(java12 & vscode-java) を追加する - mikoto2000 の日記*
  - *Eclipse Che に java12 のスタックを追加する - mikoto2000 の日記*
  - *mikoto2000/che-aarch64-simple-sample: Eclipse Che の独自スタックデモ用プロジェクト。*
- Eclipse Che が開発者向けサービスとして公開しているのでアカウントを作ればすぐに試せる
  - *Hosted Che | Eclipse Che Documentation*