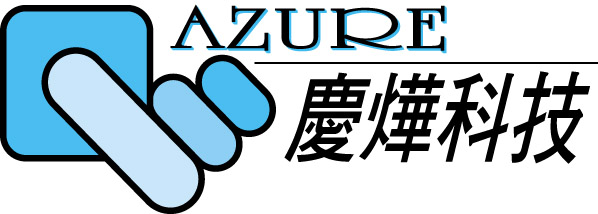
|  |
| --- |
| APConsole的標準化開發環境建置流程  *Version 1.0.0* |



|  |  |  |  |
| --- | --- | --- | --- |
| Created Date: | 2023-04-20 | By | Azure |
| Modified Date: |  | By |  |
| Azure Information Technology Co., LTD. 9F., No. 101, Sec. 3, Nanjing E. Rd., Zhongshan Dist., Taipei City 104107 , Taiwan (R.O.C.)  Tel: 886-2-2331-8866  Fax: 886-2-2370-1212 | | 慶燁科技股份有限公司  台北市中山區南京東路三段101號9樓  Tel: 886-2-2331-8866  Fax: 886-2-2370-1212 | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **版 本** | **日 期** | **異 動 說 明** | **作 者** | **審閱者** |
| 1.0.0 | 2023/04/20 | 草稿 | Daniel Lan |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

**目錄**

[一、 前言 4](#_Toc133338019)

[二、 環境架構 4](#_Toc133338020)

[三、 下載所需文件 4](#_Toc133338021)

[四、 建立編譯環境Image 4](#_Toc133338022)

[五、 建立開發環境Image 5](#_Toc133338023)

[六、 Mac與Linux 5](#_Toc133338024)

[七、 Windows 5](#_Toc133338025)

# 前言

1. 因APConsole的Product環境，大多是在RHEL系統上執行，為減少開發環境與正式環境的差異，期望開發人員是在Linux的環境開發。
2. 本案將適用Windows、Mac及Linux系統，但不同的作業系統，會有不同的IDE限制。
3. 本案將使用容器技術，務必先熟悉各項指令。

# 環境架構

專案開始進行前，一定都會規範好JDK與Node版本，依此建立CICD的編譯環境，而開發環境是基於編譯環境，再加常用的開發及網路工具。

JDK 1.8

Node.js 14

網路測試工具

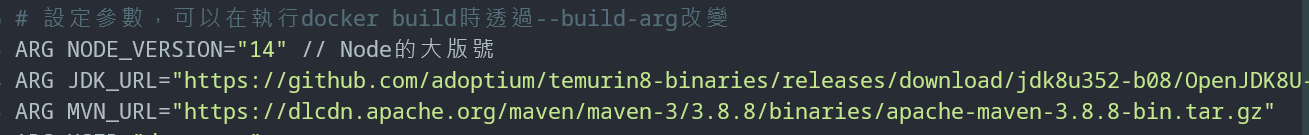
JDK 17(JDT.LS用)

# 下載所需文件

原始檔案可在Gitlab下載  
*git clone http://gitlab.cicd.local/daniellan/dev-env-standardization*

# 建立編譯環境Image

1. 編輯Dockerfile-CICD，調整JDK、Maven、Nodejs版本



1. 執行指令

*docker build -t project-builder -f Dockerfile-CICD .*

# 建立開發環境Image

執行指令

*docker build -f Dockerfile-Dev -t devEnv .*

# Mac與Linux

因Mac與Linux主要是使用devcontainer的機制，所以會有VSCode的使用限制，安裝流程如下：

# Windows