**Continuous Integration Server簡介與應用**

**前言**

在軟體開發的過程中，建置、佈署、測試及分析是不可或缺的幾個環節。過去在開發時常使用人工來處理這些項目，但這些項目通常是重複且規律的。Continuous Integration Server (以下簡稱CI) 的出現，便是為了要自動化地處理這些項目。如此一來不但節省了人力，透過CI所提供的豐富外掛，也能讓軟體開發的過程變得更方便、更順利。

**CI核心概念**

CI的精神便是重複、不斷地持續執行。CI在軟體開發流程中的角色像是個隨時存在的最佳第六人，執行建置、佈署、測試、分析的工作。

對開發人員來說，導入CI之後並沒有多大的差別，還是一樣向版本控管系統提交程式碼、撰寫單元測試等等。主要的不同是CI會使用預先撰寫好的建置腳本 (例如：Ant) 或是Maven專案中的pom.xml來建置系統，佈署環境，同時也能夠呼叫單元測試還有其他第三方的程式碼分析工具 (FindBugs, PMD, CheckStyle, …) 產生報表。增加開發者對版本控管中最新develop版本的信心。

對專案管理者來說，透過CI可以更方便地掌握專案的狀況。例如：

* 透過單元測試報表，可以看到目前測試程式的涵蓋率及通過率，了解工作的完成度及可靠度。
* 透過靜態程式碼分析報表，找出系統潛在的問題，統一開發者的程式碼風格，確保系統的品質。
* 透過自動化建置與佈署，讓開發人員不必花費時間在日常的建置與部屬，可以專注在更重要的工作。

**Jenkins簡介**

目前大家熟知的CI Server有很多種，例如Team City, CruiseControl, Jenkins, …。本篇文章將以Jenkins作為範例來展示。

Jenkins是一套由Java所開發的CI Server，而許多人可能對他的前身也十分熟悉，就是Hudson。Jenkins的介面是由Web開發，可以直接透過瀏覽器來操作。原生便可直接操作Ant Script、Maven pom.xml、Win32 script、Linux shell script、…，並透過這些腳本來完成任務。另外Jenkins更有豐富的plugin，幫助完成特殊的需求或是產生精美的報表，以下將簡單地介紹Jenkins的優點。

* 易於安裝

官網提供的安裝檔，包含三個主流系統 (Windows, Linux, Linux)。另外也提供Java Web的 .war 檔，若是電腦內已經有Java Web Container (Tomcat, Jetty, JBoss)，也能直接透過它來開啟。

* 設定簡單

CI Server往往最困擾的就是各種設定，由於Jenkins擁有豐富的plugin及Web介面，許多時候都不需要再寫任何的腳本就能完成建置的任務。

* 整合RSS/Email通知

整合通知機制，讓訂閱者可以被動地接收建置的結果，無論是取得建置成功的報表或是接收建置失敗的訊息來即時修復。

* 豐富的報表整合

Jenkins的介面是由瀏覽器來顯示，所以任何測試與靜態分析的結果都可以藉由Web介面來呈現。

* 支援分散式架構

若是公司內的程式庫內容十分龐大，可以架設多台Build Server來同時進行不同專案的建置。

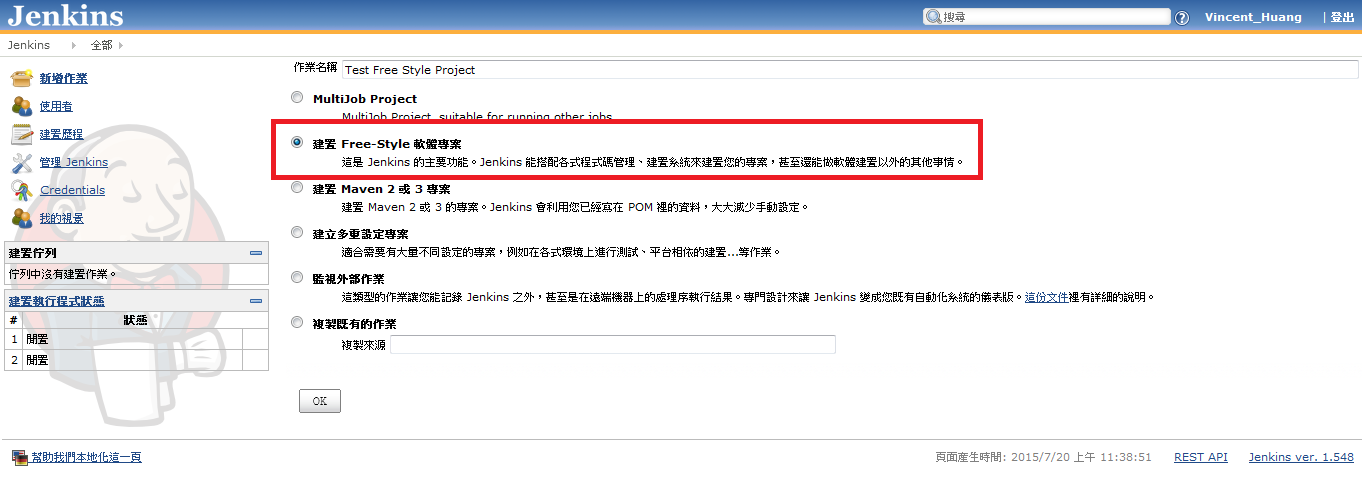
* 大量豐富的plugin

Jenkins擁有數量龐大的第三方plugins，可以根據自己的需求來選用。甚至也能撰寫自己的plugin，這方面有很大的彈性。

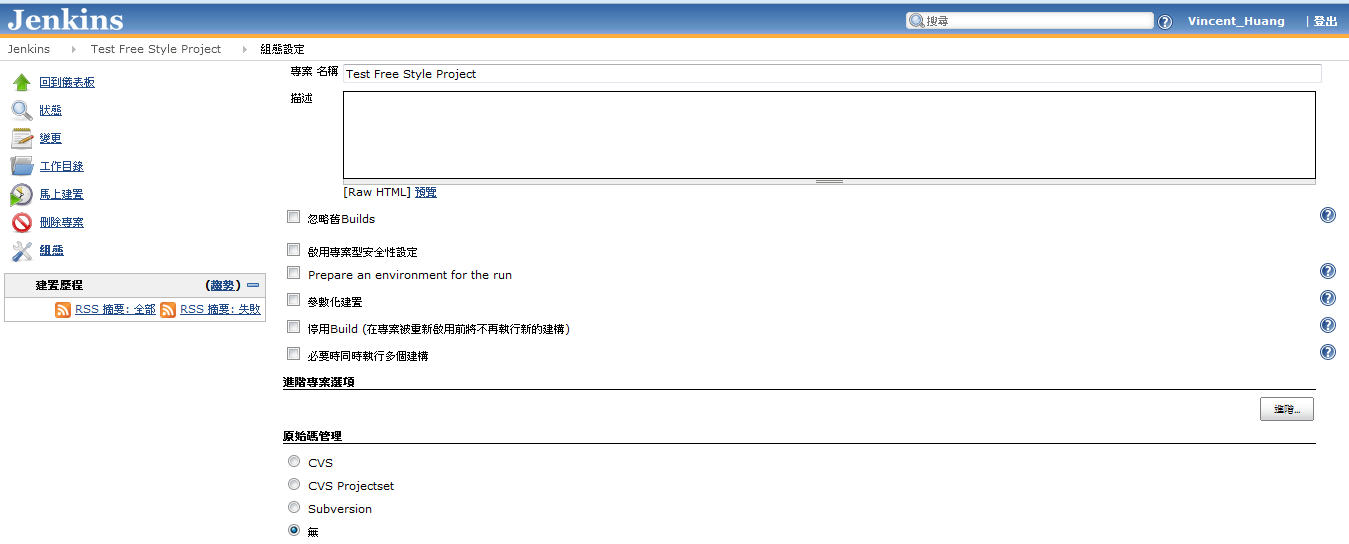
**新增建置任務**

原生的Jenkins提供幾種建置任務，在這邊主要展示建置Free-Style軟體專案

首先新增一個Free-Style軟體專案



新增完成之後，進入Test Free Style Project 的組態設定



除了有選項可供設定之外，主要需設定的是專案建置步驟，主要分為建置前步驟及建置後步驟。原生的Jenkins可加入四種步驟，包含Ant Script, Maven Build, Windows Command Line, Linux Shell Script。



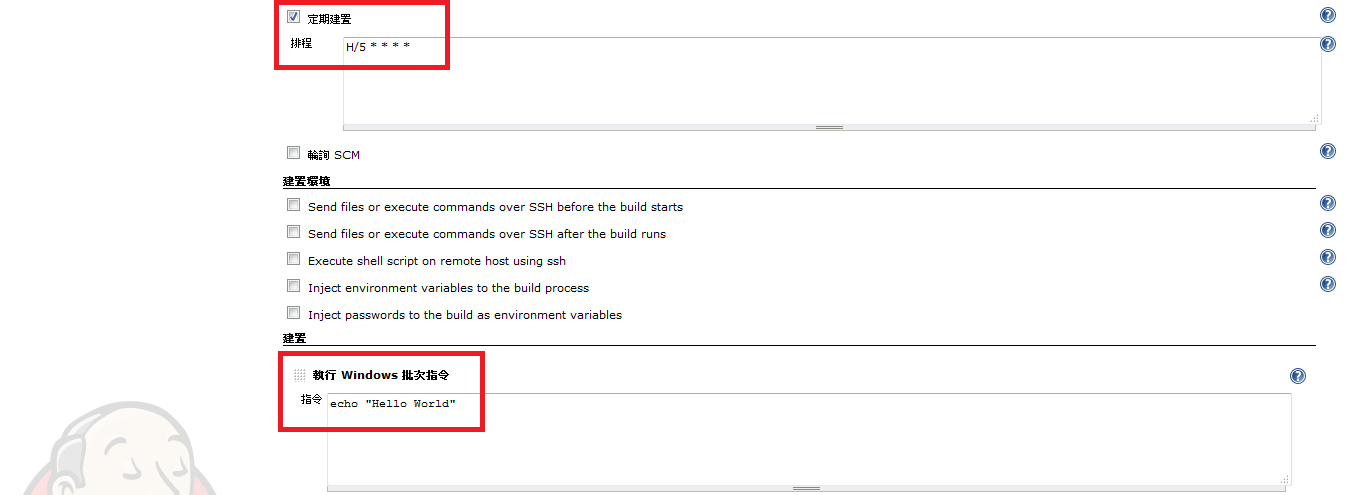
在這裡加入一個簡單的示範步驟，建立一個Windows Command Line指令

echo “Hello World”;

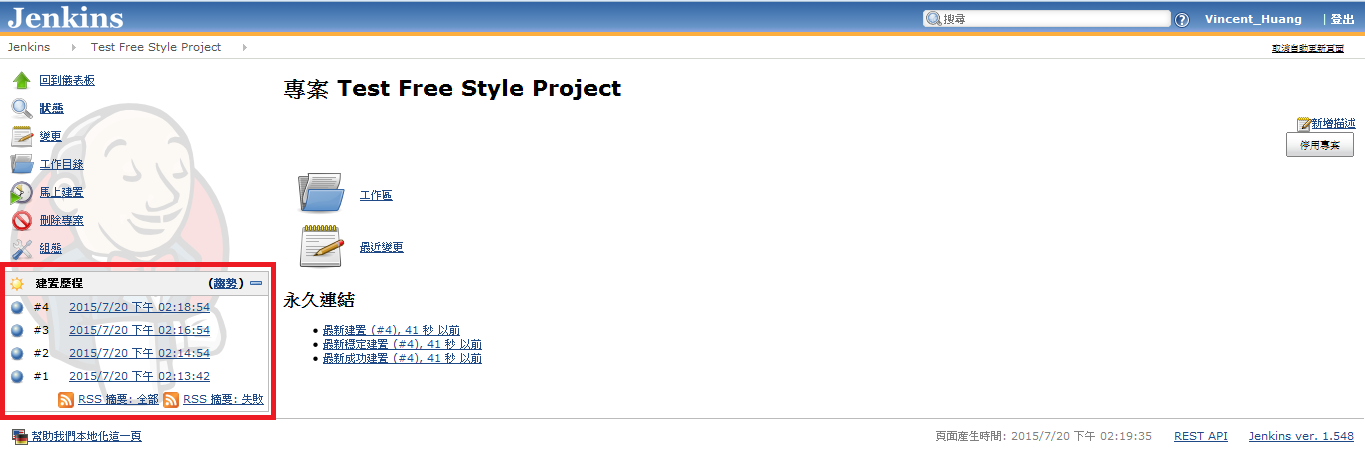
並且把定期建置的選項打勾，這裡也支援linux crontab 的表示法，輸入：

H/2 \* \* \* \*

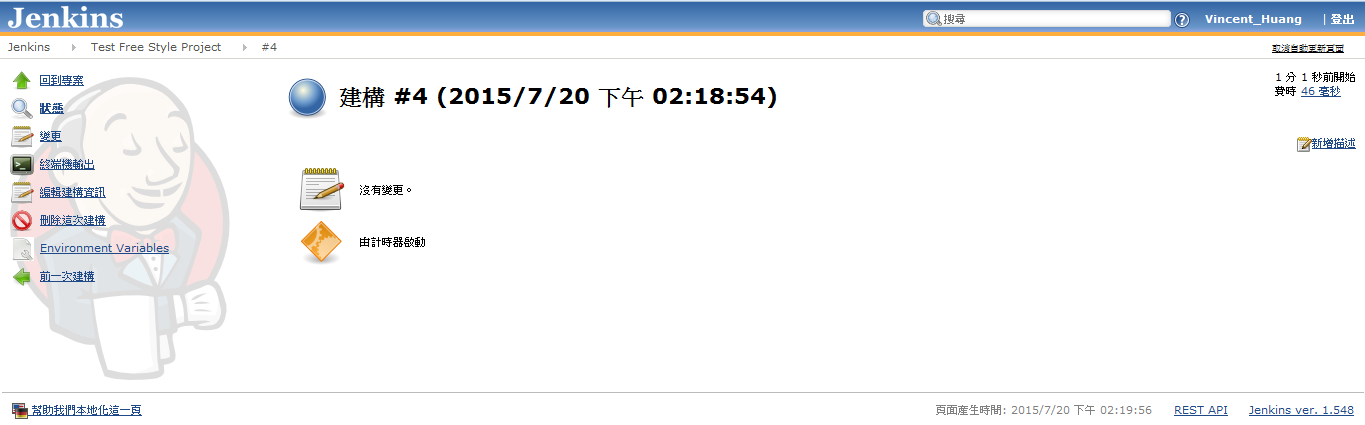
代表每兩分鐘執行一次任務



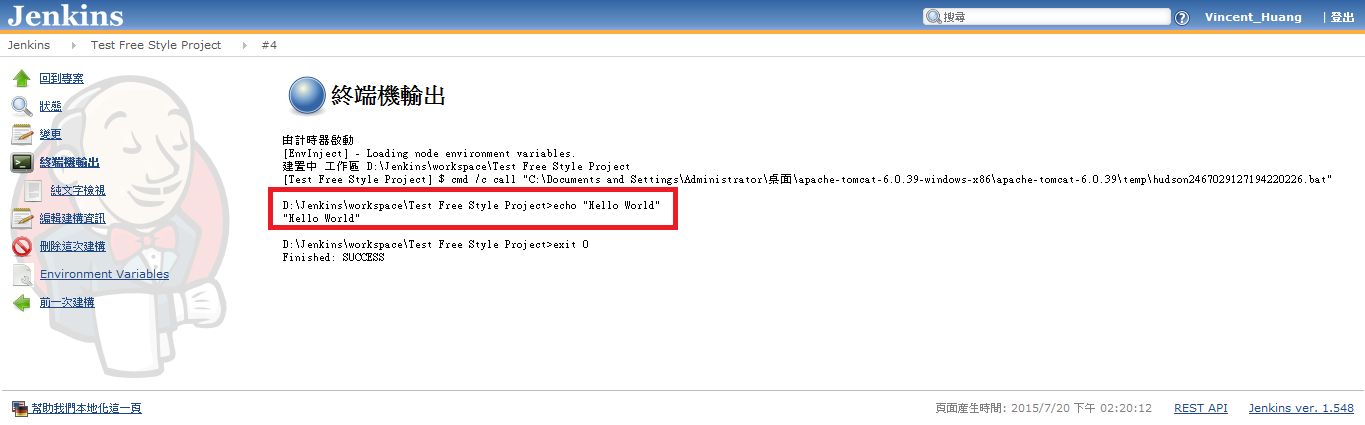
排程會每兩分鐘執行一次



進入建置結果內檢視

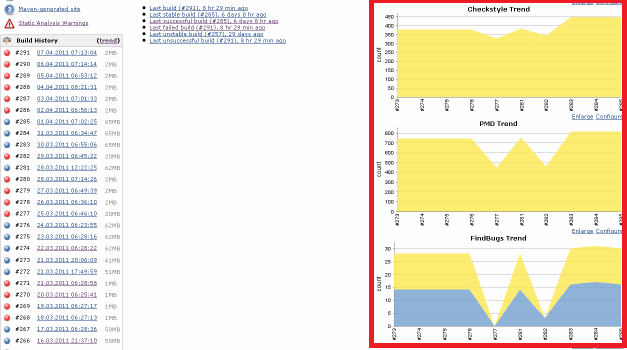


觀看終端機輸出可得到建置過程的輸出內容



**靜態程式碼分析與報表**

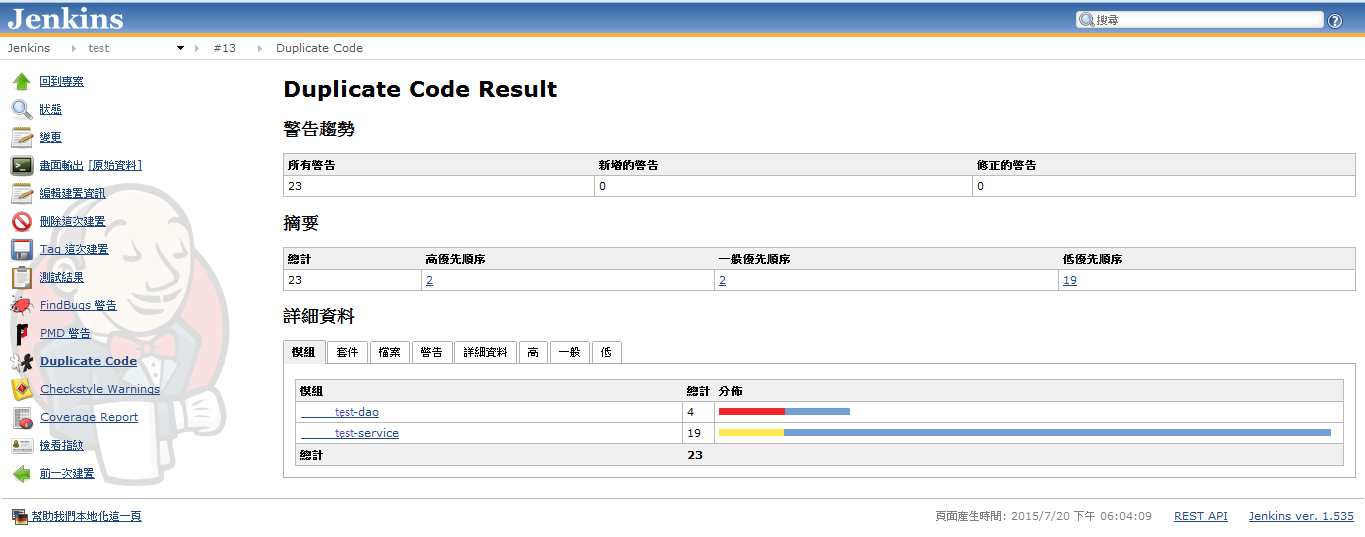
當每次專案建置完成後，便希望可以獲得一些關於此次建置的資訊。Jenkins在這個方面整合了許多已經廣為人知的靜態程式碼分析工具(如FindBugs, PMD, Duplicate Code, Checkstyle)，以plugin的形式與Jenkins來結合並透過Jenkins的Web介面來呈現報表。



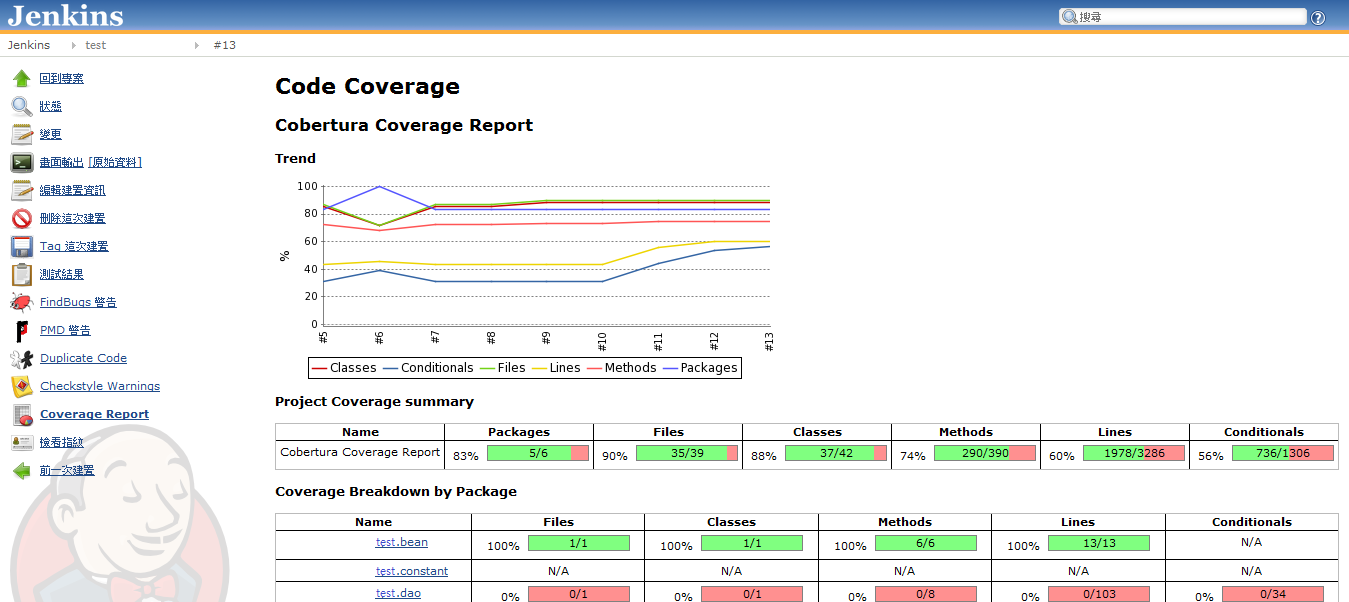












**擴充Jenkins**

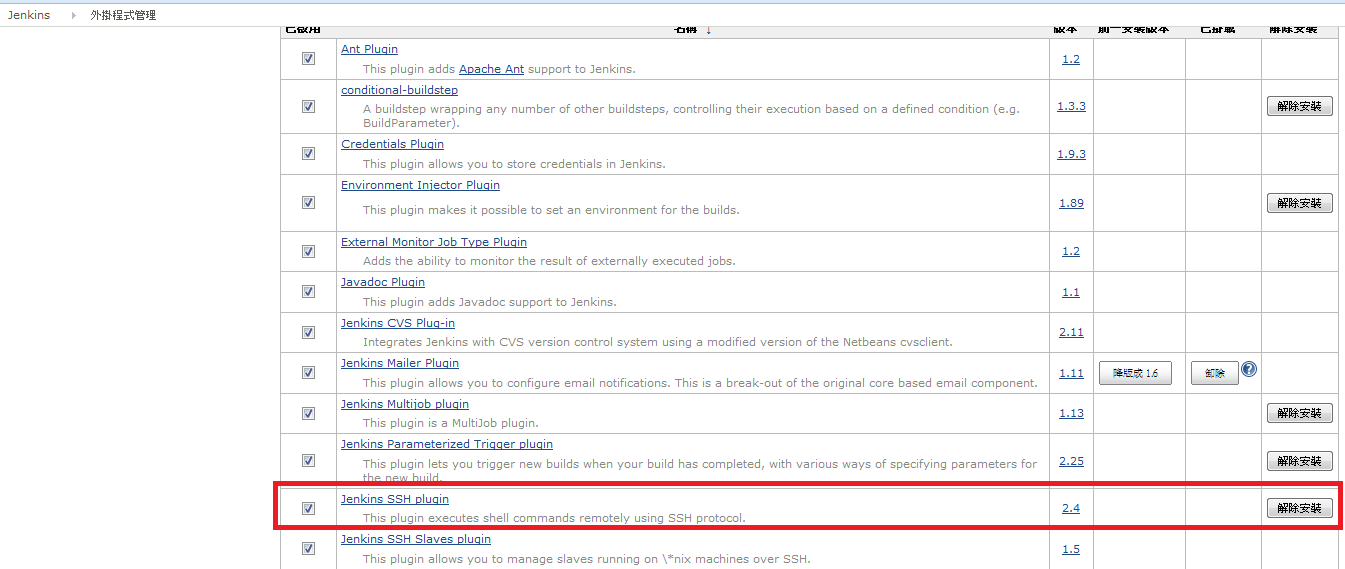
許多時候，原生的Jenkins所提供的功能可能不符合個人的需求，此時Jenkins的擴充plugin便能派上用場。







舉例來說，現在執行任務中需要使用對某台Linux主機ssh連線並且呼叫遠端的Shell Script，原生Jenkins沒有提供這個功能。此時只要安裝ssh相關的plugin，就能讓任務順利進行。





**結論**

CI Server的選擇十分地多樣化，沒有最好的選擇只有最適合的選擇，其精神遠比他的工具來的重要。更重要的是，有了CI讓開發更美好，透過CI的協助，可以讓開發人員更專注於開發。藉由CI的報表將更容易確保專案的品質，也能讓專案管理變得更輕鬆。

**參考資料**

1. <https://en.wikipedia.org/wiki/Continuous_integration>
2. <https://jenkins-ci.org>
3. <http://www.dotblogs.com.tw/hatelove/archive/2011/12/25/introducing-continuous-integration.aspx>
4. <http://blog.toright.com/posts/600/continuous-integration-%E6%8C%81%E7%BA%8C%E6%95%B4%E5%90%88%E8%88%87%E6%AF%8F%E6%97%A5%E5%BB%BA%E7%BD%AEnightly-build-hudson-%E4%BB%8B%E7%B4%B9.html>