Jenkins 入门手册 By zjl

Jenkins 入门手册

2013-08-08

写在前面的话

网络上很多大神的博客有关于 Jenkins 的文章,有点蛋疼的是。大神们往往写的很简单,像 我这种小白级人物看着像天书。我现在写下的都是我在使用 jenkins 时的一些经验。原本打 算直接在网页上写的,但是网页写着不是很方便。word 写好了,那就顺便发布一个 pdf 版 本的吧。如果文章中有错误,我会在 <u>http://www.cnblogs.com/zz0412/p/jenkins02.html</u>更正。 非常欢迎大家一起交流。

QQ 群: 172758282

邮箱: 492738140@qq.com

zjl 写于 2013-09-04

目	录
第一章	f JENKINS 是什么?1
1 JEN	KINS 的由来1
1.1	JENKINS 的目标2
第二章	f JENKINS 安裝与配置
2 JEN	KINS 安装
3 JEN	KINS 配置4
3. 1 3. 3. 3. 3. 3. 3. 2 3. 3. 3. 3. 3. 3. 3. 3. 3. 3. 3. 3. 3.	系统管理
3.3 4 JEN	监控17 NKINS 插件
4. 1 4. 2	JENKINS 插件安装

第一章 Jenkins 是什么?

Jenkins 是一个可扩展的持续集成引擎。

主要用于:

- 持续、自动地构建/测试软件项目。
- 监控一些定时执行的任务。

Jenkins 拥有的特性包括:

- 易于安装-只要把 jenkins. war 部署到 servlet 容器,不需要数据库支持。
- 易于配置-所有配置都是通过其提供的 web 界面实现。
- 集成 RSS/E-mail 通过 RSS 发布构建结果或当构建完成时通过 e-mail 通知。
- 生成 JUnit/TestNG 测试报告。
- 分布式构建支持 Jenkins 能够让多台计算机一起构建/测试。
- 文件识别: Jenkins 能够跟踪哪次构建生成哪些 jar, 哪次构建使用哪个版本的 jar 等。
- 插件支持:支持扩展插件,你可以开发适合自己团队使用的工具。

1 Jenkins 的由来

目前持续集成(CI)已成为当前许多软件开发团队在整个软件开发生命周期内侧重于保证代码质量的常见做法。它是一种实践,旨在缓和和稳固软件的构建过程。并且能够帮助您的开发团队应对如下挑战:

- **软件构建自动化**: 配置完成后, CI 系统会依照预先制定的时间表, 或 者针对某一特定事件, 对目标软件进行构建。
- 构建可持续的自动化检查: CI系统能持续地获取新增或修改后签入的 源代码,也就是说,当软件开发团队需要周期性的检查新增或修改后的 代码时,CI系统会不断确认这些新代码是否破坏了原有软件的成功构 建。这减少了开发者们在检查彼此相互依存的代码中变化情况需要花费 的时间和精力(说直接一点也是钱啊,呵呵)。
- 构建可持续的自动化测试:构建检查的扩展部分,构建后执行预先制定的一套测试规则,完成后触发通知(Email, RSS 等等)给相关的当事人。
- 生成后后续过程的自动化:当自动化检查和测试成功完成,软件构建的 周期中可能也需要一些额外的任务,诸如生成文档、打包软件、部署构 件到一个运行环境或者软件仓库。这样,构件才能更迅速地提供给用户 使用。

部署一个 CI 系统需要的最低要求是,一个可获取的源代码的仓库,一个包含构建脚本的项目。

下图概括了 CI 系统的基本结构:

Jenkins 入门手册 By zjl



图1 CI 系统的基本结构

该系统的各个组成部分是按如下顺序来发挥作用的:

- 1. 开发者检入代码到源代码仓库。
- CI系统会为每一个项目创建了一个单独的工作区。当预设或请求一次新的构建时, 它将把源代码仓库的源码存放到对应的工作区。
- 3. CI系统会在对应的工作区内执行构建过程。
- 4. (配置如果存在)构建完成后, CI系统会在一个新的构件中执行定义的一套测试。 完成后触发通知(Email, RSS 等等)给相关的当事人。
- 5. (配置如果存在)如果构建成功,这个构件会被打包并转移到一个部署目标(如应用服务器)或存储为软件仓库中的一个新版本。软件仓库可以是 CI 系统的一部分,也可以是一个外部的仓库,诸如一个文件服务器或者像 Java.net、 SourceForge 之类的网站。
- 6. CI系统通常会根据请求发起相应的操作,诸如即时构建、生成报告,或者检索一些 构建好的构件。

Jenkins 就是这么一个 CI 系统。之前叫做 Hudson。

以下是使用 Jenkins 的一些理由:

- 是所有 CI 产品中在安装和配置上最简单的。
- 基于 Web 访问,用户界面非常友好、直观和灵活,在许多情况下,还提供了 AJAX 的即时反馈。
- Jenkins 是基于 Java 开发的(如果你是一个 Java 开发人员,这是非常有用的),但 它不仅限于构建基于 Java 的软件。
- Jenkins 拥有大量的插件。这些插件极大的扩展了 Jenkins 的功能;它们都是开源的,而且它们可以直接通过 web 界面来进行安装与管理。

1.1 Jenkins 的目标

Jenkins 的主要目标是监控软件开发流程,快速显示问题。所以能保证开发人员以及相

关人员省时省力提高开发效率。

CI系统在整个开发过程中的主要作用是控制: 当系统在代码存储库中探测到修改时, 它将运行构建的任务委托给构建过程本身。如果构建失败了,那么 CI系统将通知相关人员, 然后继续监视存储库。它的角色看起来是被动的; 但它确能快速反映问题。

特别是它具有以下优点:

- Jenkins 一切配置都可以在 web 界面上完成。有些配置如 MAVEN_HOME 和 Email,只需要 配置一次,所有的项目就都能用。当然也可以通过修改 XML 进行配置。
- 支持 Maven 的模块(Module), Jenkins 对 Maven 做了优化,因此它能自动识别 Module, 每个 Module 可以配置成一个 job。相当灵活。
- 测试报告聚合,所有模块的测试报告都被聚合在一起,结果一目了然,使用其他 CI, 这几乎是件不可能完成的任务。
- 构件指纹(artifact fingerprint),每次 build 的结果构件都被很好的自动管理,无需 任何配置就可以方便的浏览下载。

第二章 Jenkins 安装与配置

2 Jenkins 安装

在最简单的情况下, Jenkins 只需要两个步骤:

1.下载最新的版本 (一个 WAR 文件)。Jenkins 官方网址: http://Jenkins-ci.org/

2.运行 java -jar jenkins.war

注意: Jenkins 需要运行 Java 5 以及以上的版本。

还有一种安装方式就是将下载的 war 包文件部署到 servlet 容器,然后启动容器,在浏 览器的 URL 地址栏中输入类似 <u>http://localhost:8080/jenkins/</u>这样的地址即可。下图是安装成 功后的界面(使用的是 Linux+Tomcat6+Java6 环境):



图2 Jenkins 主界面

3 Jenkins 配置

写在配置前的一些话: Jenkins 的配置不可能全部都说到的,大部分配置是有英文说明的, 点击输入框后面的问号就可以看见了。英文不会用翻译工具,连蒙带猜的,多测试几次,你 就懂了。

3.1 系统管理

在已运行的 Jenkins 主页中,点击左侧的系统管理进入如下界面:

系统管理

<u>^</u>	Your container doesn't use UTF-8 to decode URLs. If you use non-ASCII characters as a job name etc, this will cause problems. See <u>Containers</u> and <u>Tomcat i18n</u> for more details. New version of Jenkins (1.518.JENKINS-14362-izlib) is available for download (changelog).					
Λ	Unsecure Consider	d Jenkins allows anyone on the network to launch processes on your behalf. at least enabling authentication to discourage misuse.	Setup Security	Dismiss		
	X	<u>系统设置</u> 全局设置&路径				
		<u>Configure Global Security</u> Secure Jenkins; define who is allowed to access/use the system.				
	Z	读取设置 放弃当前内存中所有的设置信息并从配置文件中重新读取 仅用于当您手动修改配置文件时重新读取	设置。			
	-	<mark>管理插件</mark> 添加、删除、禁用或启用Jenkins功能扩展插件。 (updates available)				
		<u>系统信息</u> 显示系统环境信息以帮助解决问题。				
	Ē,	<u>System Log</u> 系统日志从java.util.logging捕获Jenkins相关的日志信息。				
		<u>负载经计</u> 检查您的资源利用情况,看看是否需要更多的计算机来帮助您构建。				
		<u>Jenkins CLI</u> 从您命令行或脚本访问或管理您的Jenkins。				
	0000000	<u>脚本命令行</u> 执行用于管理或故障探测或诊断的任意脚本命令。				
		团0 1 1. 石仿做四				

图3 Jenkins 系统管理

3.1.1 提示信息

Ps: 版本不同提示的消息有可能不同

3.1.1.1 Utf-8 编码

Your container doesn't use UTF-8 to decode URLs. If you use non-ASCII characters as a job name etc, this will cause problems. See <u>Containers</u> and <u>Tomcat i18n</u> for more details.

Jenkins 建议在 tomcat 中使用 utf-8 编码, 配置 tomcat 下 conf 目录的 server.xml 文件

```
<Connector port="8080" URIEncoding="UTF-8" protocol="HTTP/1.1"
connectionTimeout="20000"
redirectPort="8443" />
```

图4 URIEncoding 编码设置

Ps: 如果 Job 的控制台中文输出乱码,请将 URIEncoding="utf-8"更改为 useBodyEncodingForURI="true"

3.1.1.2 新的版本

New version of Jenkins (1.518.JENKINS-14362-jzlib) is available for <u>download</u> (<u>changelog</u>).

提示有新的版本可以下载了,喜欢更新的点击 download 去下载吧!

3.1.1.3 安全设置

Unsecured Jenkins allows anyone on the network to launch processes on your behalf.
 Consider at least enabling authentication to discourage misuse.

图5 安全提示消息

詹金斯允许网络上的任何人代表您启动进程。考虑至少启用身份验证来阻止滥用。点击 Dismiss 忽略该消息,点击 Setup Security 进入设置界面.详细设置请参考 Configure Global Security(安全设置)章节

3.1.2 系统设置

在已运行的 Jenkins 主页中,点击左侧的系统管理—>系统设置进入如下界面:

主目录	/root/.jenkins	0
	高級	
系统消息		?
	[Raw HTML] <u>预览</u>	
执行者数里	2	
生成前等待时间	5	2
SCM签出重试次数	0	

图6 系统设置界面

Setup Security

3.1.2.1 JDK、Maven、Ant 配置

配置一个 JDK、Ant、Maven 实例,请在每一节下面单击 Add(新增) 按钮,这里将添加实例 的名称和绝对地址。下图描述了这两个部分。

JDK		
JDK 安装	JDK 别名 jdk1.6.0_23	
	JAVA_HOME //usr/java/jdk1.6.0_23/	
	□ 自动安装	Ø
		册除 JDK
	新增JDK	
	系统下 JDK 安装列表	

图7 JDK 配置界面

JDK 别名:给你看的,随便你自己,叫阿猫阿狗都可以 JAVA_HOME:这个是本机 JDK 的安装路径(错误的路径会有红字提示你的) 自动安装:不推荐这个选项 后面 Ant 与 Maven 的配置是一样的,JDK 去 oracle 官网下载,Ant 与 Maven 去 apache 官网下载 Ps:每个文本框后面都有个问号,点击问号就会出现帮助信息

3.1.2.2 邮件通知配置

3.1.2.2.1 配置发件人地址

Jenkins Location			
Jenkins URL	http://192.		
System Admin e-mail address	jsbjenkin ter and n.cn		

图8 发件人地址配置界面

System Admin e-mail address: Jenkins 邮件发送地址,如果你这个没有配置,等着发邮件的时候报 错吧,当时我也是这儿没有配置,郁闷了我一周的时间。①___①b 汗

3.1.2.2.2 配置邮件通知

邮件通知			
SMTP服务器	www		2
用户默认邮件后缀	@ cn		2
☑ 使用SMTP认证		(2
用户名	isbjenkins		
密码			
使用SSL协议		(0
SMTP端口			2
Reply-To Address			
字符集	UTF-8		
🗹 通过发送测试邮件测试配置			
Test e-mail recipient	testxxx		
	Email was successfully sent	Test configuration	

图9 邮件通知

这个就非常的简单了,根据的的邮箱提供者的参数配置就行了。 Ps:小技巧:用户默认邮件后缀配置了后,以后你填写邮件地址只需要@之前的就行了

3.1.2.3 Subversion 配置

Subversion

Subversion Workspace Version	1.7	
Exclusion revprop name		

- Validate repository URLs up to the first variable name
- 🗹 Update default Subversion credentials cache after successful authentication

图10 Subversion 配置

Subversion Workspace Version: Subversion 的版本号,选择你对应的版本号就行了

3.1.3 Configure Global Security(安全设置)

在已运行的 Jenkins 主页中,点击左侧的**系统管理—>Configure Global Security** 进入如下界面:



☑ 启用安全 JNLP节点代理的TCP端口	◎ 指定端口: 🔄 💿 随机选取 💿 禁用
Markup Formatter	Raw HTML
	将文本作为HTML并且不加任何翻译
访问控制	安全域
	◎ Jenkins专有用户数据库
	☑ 允许用户注册
	C LDAP
	◎ Servlet容器代理
	◎ Unix用户/组数据库
	授权策略
	◎ 任何用户可以做任何事(没有任何限制)

图11 安全设置界面

设置如上图,保存后**系统管理**中就出现**管理用户**的选项。页面右上角也会出现**登录/注册**的 选项。

3.1.4 管理用户设置

在右上角点击注册

Sign up

用户名:	test
密码:	••••
确认密码 <mark>:</mark>	••••
全名:	测试用户
电子邮件地址 <mark>:</mark>	test@test.com
0	

Sign up

图12 注册用户界面

点击 sign up 按钮,提示**你现在已经登录.返回首页**. 登录后和匿名账号看到的首页有几点不同,如下图红框所示:

Jenkins	◎ 搜索	() t	est 退出
Jenkins >		ENABLE AUTO	REFRESH
 ☆ 近10b ☆ 査重用户 ☆ 査重用户 ☆ 任务历史 ※ 系統管理 秋的视图 ▲ 竹v Views 任务队列 当前队列没有物理任务 		用户名	☑添加说明
任务状态 # 状态 1 空闲 2 空闲			
12. 帮助我们本地化当前页	页面生成于: 2013-6-18 18:38:02	REST API Jenkin	s ver. 1.518

图13 用户登录界面

3.1.5 管理插件设置

建议先阅读 <u>Jenkins 插件</u>章节,在回来安装如下所示的插件。这个插件将生成的构件(war 或者 ear)部署到主流的服务器上。

插件名称: <u>Deploy Plugin</u>

插件介绍: This plugin takes a war/ear file and deploys that to a running remote application server at the end of a build

3.2 项目构建设置

3.2.1 构建自由风格的 Job

3.2.1.1 新建自由风格构建任务

在已运行的 Jenkins 主页中,点击左侧的新建 Job 进入如下界面:

1±	穷名称	Ant_test
٩	构建-	ー 个自由风格的软件项目 这是Jenkins的主要功能.Jenkins将会结合任何SCM和使用任何构建系统来构建你的项目,甚至可以使用软件构建以外的系统.
0	构建-	ー 个maven2/3项目 构建一个maven2/3项目.Jenkins利用你的POM文件.这样可以大大臧轻构建配置.
0	构建-	ー 个多配置项目 适用于多配置项目,例如多环境则试,平台指定构建,等等,
0	监控-	ー 个外部的任务 这个类型的任务允许你记录执行在外部Jenkins的任务,任务甚至运行在远程机器上.这可以让Jenkins作为你所有自动构建系统的控制面板.参阅 <u>这个文档查看详细内容</u> .
0	拷贝i	已存在任务 要复制的任务名称
ſ	ок	

图14 新建 Job 界面

这时,需要为新的构建任务指定一个名称。(这里输入的任务名称为: Ant_test)这里有几种

的任务类型可供选择,鉴于初步介绍,先选择构建一个自由风格的软件项目。对于其他的类型, 经常使用的是拷贝已存在任务;这主要为了能在现有的任务基础上新建任务。点击 OK 按钮,

3.2.1.2 构建任务配置

3.2.1.2.1 源码管理配置

演示是使用 Subversion 的链接,在 Repository URL 中输入你的项目链接,如果没有权限则会提示 如下图:

源码管理				
CVS				
CVS Projectset				
None				
Subversion				_
Modules	Repository URL	https://192	iness_client/	Ø
		Unable to access		-
		https://	usiness_client/	
		Svn: E200015: OPTIONS (svn/generation)	client failed (show details)	
		(Maybe you need to <u>enter credential</u> ?)	cient funea (Snow details)	
	Local module directory (option	nal) .		0
	Repository depth option	infinity 💌		0
	Ignore externals option			0
			Add more locations	
Check-out Strategy	^y Use 'svn update' as much as p	ossible		-
	Use 'svn update' whenever po when a new build starts.	ossible, making the build faster. But this causes the	artifacts from the previous build to rem	ain
源码库浏览器	(自动)			-
				_
			目初	2

图15 Subversion 配置界面

点击 enter credential 输入用户名和密码(我猜大家一般都是使用的用户名和密码登陆的)

Enter the authentication information needed to connect to the Subversion repository. This information will be stored in Jenkins.

	nups.//192	clienty	
Osername/pass	word authentication		
User name	test		
Password			
	••••		
SSH public key	authentication (svn+ssh)		
HTTPS client cer	tificate		
ОК			

图16 Subversion 权限认证界面

Ps: svn 的用户名和密码设置了是没有办法在 web 界面修改的。如果要修改则先去 Jenkins 目录删除 hudson.scm.SubversionSCM.xml 文件(点到为止,毕竟这只是入门教程)

3.2.1.2.2 构建触发器

在其他项目构建完成后才执行构建:指定的项目完成构建后,触发此项目的构建。 Poll SCM:这是 CI 系统中常见的选项。当您选择此选项,您可以指定一个定时作业 表达式来定义 Jenkins 每隔多久检查一下您源代码仓库的变化。如果发现变化,就执行一次构建。例如,表达式中填写 0,15,30,45 * * * *将使 Jenkins 每隔 15 分钟就检查一次您源码仓库的变化。

Build periodically:此选项仅仅通知 Jenkins 按指定的频率对项目进行构建,而不管 SCM 是否有变化。如果想在这个 Job 中运行一些测试用例的话,它就很有帮助。

3.2.1.2.3 Ant 构建配置

因为我的项目是用 ant 脚本实现的编译和打包,所以我选择的是 Invoke Ant, Ant Version 选择你 Ant 配置的那个名字,注意不要选择 default 喔,那个选择了没有用。

	4建							
	增加构建	步骤▼						
-	Execu Execu	te Wind te shell	lows batc	h command				
\leq	Invok	e Ant						
	Invok	e top-le	vel Maver	targets				
<u>构建</u>	nvoke Ant				 	 		0
<u>构建</u> Ⅲ I Ant	nvoke Ant : Version a	nt						0
<u>拘建</u> Ⅲ I Ant Tar	nvoke Ant : Version a gets a	nt efault nt						 •

图17 Ant 构建配置界面

如果你的构建脚本 build.xml 不在 workspace 根目录、或者说你的构建脚本不叫 build.xml。那么需要 在高级里设置 Build File 选项的路径,指明你的脚本。注意:是相对路径 部署请参考:war文件部署章节

3.2.2 构建 Maven 风格的 Job

3.2.2.1 新建 Maven 构建任务

£务	5名称	maven_test
) ;	构建-	- 个自由风格的软件项目 这是Jenkins的主要功能.Jenkins将会结合任何SCM和使用任何构建系统来构建你的项目,甚至可以使用软件构建以外的系统.
) ;	构建-	ー 个maven2/3项目 构建一个maven2/3项目.Jenkins利用你的POM文件,这样可以大大减轻构建配置.
);	构建-	- 个多配置项目 适用于多配置项目,例如多环境测试,平台指定构建,等等.
	监控-	- 个外部的任务 这个类型的任务允许你记录执行在外部Jenkins的任务,任务甚至运行在远程机器上,这可以让Jenkins作为你所有自动构建系统的控制面板。 参阅 <u>这个文档查看详细内容</u> 。
	拷贝已	2存在任务 要复制的任务名称
(ок	
		H务名称 内建一 内建一 内建一 均建一 均建一 均建一 均建一 均建一 5 5 5 6 7 <

图18 新建 Job 界面

这时,需要为新的构建任务指定一个名称。(这里输入的任务名称为: maven_test)这里有几种的任务类型可供选择,鉴于初步介绍,先选择**构建一个 maven2/3 项目**。对于其他的类型,经常使用的是**拷贝已存在任务**;这主要为了能在现有的任务基础上新建任务。点击 OK 按钮,

3.2.2.2 构建任务配置

Project名称 Ant_test	
描述	
【Raw HTML】 预览 ■ 丢弃旧的构建	0
■ 参数化构建过程	0
🥅 停止构建(直到允许项目构建,否则不能进行新的构建)	2
🔲 在必要的时候并发构建	0
高级项目选项	
高级	

源码管理

CVS
 CVS Projectset

None

Subversion

图19 构建任务配置界面

3.2.2.2.1 源码管理配置

演示是使用 Subversion 的链接,在 Repository URL 中输入你的项目链接,如果没有权限则会提示 如下图:

源码管理			
CVS			
CVS Projectset			
None			
Subversion			
Modules	Repository URL	https://192	iness dient/
	l	linable to access	
		https://	usiness_client/
		C : svn: E200015: OPTIONS	
		/svn/d (Maybe you need to enter credential?)	client failed <u>(show details)</u>
	Local module directory (optional	· · · · · · · · · · · · · · · · · · ·	0
		•	¥
	Repository depth option	infinity 💌	0
	Ignore externals option		®
			Add more locations
Check-out Strate	Use 'svn update' as much as poss	sible	
	Use 'svn update' whenever possi when a new build starts.	ble, making the build faster. But this causes the	artifacts from the previous build to remain
源码库浏览器	(自动)		▼ 0
			高级
	反	90 Subversion 配異更面	
	a de la companya de la	120 Subversion 配直外回	
$\vdash \perp$			

点击 enter credential 输入用户名和密码(我猜大家一般都是使用的用户名和密码登陆的)

Subversion Authentication

Enter the authentication information needed to connect to the Subversion repository. This information will be stored in Jenkins.

	Repository URL	https://192
0	Username/pass	sword authentication
	User name	test
	Password	••••
C	SSH public key	authentication (svn+ssh)
C	HTTPS client cer	tificate
	ок	

图21 Subversion 权限认证界面

Ps: svn 的用户名和密码设置了是没有办法在 web 界面修改的。如果要修改则先去 Jenkins 目录删除 hudson.scm.SubversionSCM.xml 文件(点到为止,毕竟这只是入门教程)

3.2.2.2.2 构建触发器

在其他项目构建完成后才执行构建:指定的项目完成构建后,触发此项目的构建。

Pol1 SCM: 这是 CI 系统中常见的选项。当您选择此选项,您可以指定一个定时作业 表达式来定义 Jenkins 每隔多久检查一下您源代码仓库的变化。如果发现变化,就执行一次 构建。例如,表达式中填写 0,15,30,45 * * * *将使 Jenkins 每隔 15 分钟就检查一次您源码仓 库的变化。

Build periodically:此选项仅仅通知 Jenkins 按指定的频率对项目进行构建,而不管 SCM 是否有变化。如果想在这个 Job 中运行一些测试用例的话,它就很有帮助。

3.2.2.3 Maven 构建设置

Pre Steps		
Add pre-build step 🕶		
Build		
Root POM	pom.xml	0
Goals and options	clean install	0
		高级
Post Steps		
	$^{\odot}$ Run only if build succeeds $^{\odot}$ Run only if build succeeds or is unstable $^{\odot}$ Run only if build result	un
Add post-build step 🗸	Should the post-build steps run only for successful builds, etc.	

图22 Maven 构建配置界面

2013-08-22 补充 Goals and options : clean install -Dmaven.test.skip=true #加入了跳过测试的代码

Root POM:填写你项目的 pom.xml 文件的位置,注意:是相对位置,如果该文件不存在,会有红色字提示。

部署请参考:war文件部署章节

3.2.2.2.4 构建 maven 项目的心得

使用 Jenkins 构建 maven 项目的一点小心得:

maven 项目的构建是比较麻烦的,如果你的项目是下图这种结构。那么恭喜你!你新建一个 job 就可以 了,因为只有一个根。如果你的 svn 地址是:https://192.xxx/Pe_Project/root-pom,那么 Root POM 只需要保持默认就行了,因为 Jenkins 可以再 workspace 目录下面找到 pom.xml 文件 如果你的 svn 地址是: https://192.xxx/Pe_Project,那么 Root POM 需要指定为 root-pom/pom.xml,因为

Jenkins 可以再 workspace/root-pom 目录下面找到 pom.xml 文件

	.settings	70	2013/8/6 16:3
	📗 if-base	76	2013/8/7 18:0
.settings	📗 if-framework	78	2013/8/7 18:0
root-pom	🔰 if-util	77	2013/8/7 18:0
.settings	🔒 pe-acct-api	70	2013/8/6 16:3
🚺 if-base	📗 pe-acct-api-impl	70	2013/8/6 16:3
🐌 if-framework	📗 pe-api	70	2013/8/6 16:3
🐌 if-util	📗 pe-api-impl	70	2013/8/6 16:3
퉬 pe-acct-api	📗 pe-quartz	70	2013/8/6 16:3
🎍 pe-acct-api-impl	📗 pe-war	70	2013/8/6 16:3
🎍 pe-api	Servers	70	2013/8/6 16:3
📗 pe-api-impl	.project	.project 70	388 字节 2013/8/6 16:3
📗 pe-quartz	.project.bak	.bak 70	393 字节 2013/8/6 16:3
pe-war	pom.xml	.xml 75	4.32 KB 2013/8/7 17:3
Servers	=		http

图23 Maven 项目结构界面 1

上面这种方法打包的时候非常简单,但是用 eclipse 开发的时候你就不右键 run as —>tomca 启动了,如果你想使用这种方式,将 tomcat 换成 jetty 即可。

如果你的项目是下图这种结构,那么非常悲剧的告诉你,你要建立好几个 job 来构建这一个

项目,因为这个项目有4个根。

URL: O https://	oss/trunk		
https://tooss branches tags trunk boss-jar if-root validateCode	文件 boss-jar if-root root-pom validateCode context.xml giftcard 20130321 1717.sql	扩展名 .xml .sql	. 1

图24 Maven 项目结构界面 2

上面这种方法打包的时候比较麻烦,但是用 eclipse 开发的时候你就可以使用右键 run as —>tomca 启动了

3.2.3 邮件通知设置

<u>构建后操作</u>

Add post-build action 💌

图25 构建后操作界面

选择 Add post-build action, 然后选择 E-mail Notification, 如下图:

🖩 E-mail N	lotification	0
Recipients		
	Whitespace-separated list of recipient addresses. May reference build parameters like \$PARAM. E-mail will be sent when a build fails, becomes unstable or returns to stable.	
	🗹 每次不稳定的构建都发送邮件通知	
	🔲 单独发送邮件给对构建造成不良影响的责任人	0
	明時	
Add post-bui	Id action 👻	

图26 收件人列表界面

在 Recipients 中输入收件人邮件地址,如果用多个收件人用","英文逗号隔开

3.2.4 War 文件部署设置

首先你必须安装好 <u>Deploy Plugin</u>插件,然后在 tomcat 的 conf 目录配置 tomcat-users.xml 文件,在<tomcat-users>节点里添加如下内容:<role rolename="manager-gui"/>

<role rolename="manager-script"/>

<role rolename="manager-jmx"/>

<role rolename="manager-status"/>

<user username="username" password="password"

 $roles = "manager-gui, manager-script, manager-jmx, manager-status" /\!\!>$

引号里的 username 和 password 可以随便替换,待会要用的。

好了,回到 Jenkins 项目配置页面:

<u>构建后操作</u>

Add post-build action -

图27 构建后操作界面

选择 Add post-build action, 然后选择 Deploy war/ear to a container, 如下图: **物建后操作**

Deploy war/e	ear to a container		
WAR/EAR files	target/test.war	0	
Context path	ofCard	0	
Container	Tomcat 6.x	•	
	Manager user name	username	
	Manager password	•••••	
	Tomcat URL	http://192.168.x.x:8080/	
Deploy on failur	e 🕅		

图28 远程部署配置界面

WAR/EAR files: war 文件的存放位置,如: target/test.war 注意:相对路径, target前是没有/的。 Context path:访问时需要输入的内容,如<u>ofCard</u>访问时如下: <u>http://192.168.x.x:8080/ofCard/</u> 如果为空,默认是 war 包的名字。 Container:选择你的 web 容器,如 tomca 6.x Manager user name:填入 tomcat-users.xml 配置的 username 内容

Manager password: 填入 tomcat-users.xml 配置的 password 内容

Tomcat URL: 填入 <u>http://192.168.x.x:8080/</u>

Deploy on failure: 构建失败依然部署, 一般不选择

注意:虽然这种部署方法可能会导致 tomcat 加载时出现卡死的现象。但是也是最简单的部署方式。如果 卡死了重启下就好了,将 tomcat 的 java 内存参数调高可以解决这个问题。

最后不要忘记点击保存喔。

好了! 到此一个项目的获取源码, 打包, 远程部署, 邮件通知就完成了。

删除

3.3 监控

当任务一旦运行,您将会看到这个任务正在队列中的仪表板和当前工作主页上运行。这 两种显示如下。



图29 主页监控(左),项目监控(右)

一旦构建完成后,完成后的任务将会有三个地方进行显示。

你可以在 Jenkins 的控制面板上看到它,如下图。

S	w	Name ↓	上次成功	上次失败	上次持续时间	
		Ant test	18分- <u>#16</u>	20 分 - <u>#15</u>	22 秒	\bigotimes
		图:	30 主页项目界面			

在上面展示的截图中,您将注意到有两个图标描述当前作业的状态。S 栏目代表着"最新构建状态", W 栏目代表着"构建稳定性"。Jenkins 使用这两个概念来介绍一个作业的总体状况:

构建状态:下图中分级符号概述了一个 Job 新近一次构建会产生的四种可能的状态:

Successful:完成构建,且被认为是稳定的。

Unstable:完成构建,但被认为不稳定。

Failed:构建失败。

Disabled:构建已禁用。

Unstable	Broken	Disabled
	Unstable	Unstable Broken

图31 构建状态界面

构建稳定性: 当一个 Job 中构建己完成并生成了一个未发布的目标构件,如果您准备评估此 次构建的稳定性, Jenkins 会基于一些后处理器任务为构建发布一个稳健指数(从 0-100), 这些任务一般以插件的方式实现。它们可能包括单元测试(JUnit)、覆盖率(Cobertura)和 静态代码分析(FindBugs)。分数越高,表明构建越稳定。下图中分级符号概述了稳定性的评 分范围。任何构建作业的状态(总分 100)低于 80 分就是不稳定的。



图32 构建稳定性界面

你也可以在当前 Job 主界面上看到它,如下图左下部分

	Project Ant test
<u> </u>	
🔁 变更集	
🍯 I作区	
🔊 立即构建	工作区
S 删除 Project	
💥 配置	▶ 最近变更集
Favorite	
Build History (約期)	永久连接
#16_2013-8-12_14:24:257KP	◆ Last build(#16),19 分前
#15 2013-8-12 14:22:10 11KB	 Last stable build(#16),19 分前 Last successful build(#16),19 分前
#14 2013-8-12 14:19:14 15KB	Last failed build(#15).21 分前 Last unsuccessful build(#15) 21 分前
#13 2013-8-12 14:13:13 44KB	
#12 2013-8-2 13:30:29 38KB	
#11 2013-7-17 16:25:43 37KB	
#10 2013-7-17 16:16:07 З7КВ	
• #9 <u>2013-7-17 16:15:51</u> 37KB	
• #8 <u>2013-7-17 14:18:04</u> 37KB	
• #7 <u>2013-7-17 14:16:24</u> 37KB	
#6 <u>2013-7-17 10:40:16</u> 37KB	
• #5 <u>2013-7-17 9:49:43</u> 37KB	
4 2013-7-17 9:46:12 37KB	
#2 2013-7-17 9:44:08 27/P	
#1 2013-7-16 16:20:37 80KB	
■ 0000 ■ RSS 全部 图 RSS 失败	

图33 项目主界面

当前作业主页上还包含了一些有趣的条目。左侧栏的链接主要控制 Job 的配置、删除作业、构建作业。右边部分的链接指向最新的项目报告和构件。

通过点击构建历史(Build History)中某个具体的构建链接,您就能跳转到 Jenkins 为这 个构建实例而创建的构建主页上。如下图



📲 帮助我们本地化当前页

页面生成于: 2013-8-12 14:45:30 <u>REST API</u> <u>Jenkins ver. 1.522</u>

图34 构建历史界面

如果你想通过视图输出界面来监控当前任务的进展情况。你可以单击 Console Output (控制 **台输出)**。如果工作已完成,这将显示构建脚本产生的静态输出,如果作业仍然在运行中, Jenkins 将不断刷新网页的内容,以便您可以看到它运行时的输出。如下图:



图35 控制台输出界面

4 Jenkins 插件

从 Jenkins 现有的功能扩展或开发者们为 Jenkins 提供的新功能都可以称之为 Jenkins 插件。有些插件可以无缝添加到您的构建过程,而其它,诸如除 CVS 和 Subversion 的 SCM 插件则需要源代码控制系统的支持。

4.1 Jenkins 插件安装

Jenkins 插件管理器允许您安装新的插件,和更新您 Jenkins 服务器上的插件。管理者 将连接到联机资料库,检索可用的和己更新的插件。如果您的 Jenkins 服务器 无法直接连 接到外部资源,您可以从 Jenkins 网站上下载。

在已运行的 Jenkins 主页中,点击左侧的系统管理—>管理插件进入如下界面:

と装 👃	名称	版本	已安装
	LDAP Plugin	1.6	1 5
	Security realm based on LDAP authentication.		
	Credentials Plugin	16	1.4
	This plugin allows you to store credentials in Jenkins.	1.0	1.4
	SSH Credentials Plugin		
	This plugin allows you to store SSH credentials in Jenkins.	1.3	0.4
	Warning: New version not compatible with installed version. Jobs using this plugin may need to be reconfigured.		
	Token Macro Plugin	1.0.1	17
	This plugin adds reusable macro expansion capability for other plugins to use	1.0.1	1./
	Maven Project Plugin		
	Special project type for Maven projects.	1.526	1.522
	Warning: This plugin is built for Jenkins 1.526 or newer. It may or may not work in your Jenkins.		
	Static Code Analysis Plug-ins	1.51	1 50
	This plug-in provides utilities for the static code analysis plug-ins.		1.50
	CVS Plugin		2.0
	This bundled plugin integrates Jenkins with CVS version control system.	2.9	2.0
	SSH Slaves plugin		
	This plugin allows you to manage slaves running on *nix machines over SSH.	1.2	0.27
	Warning: New version not compatible with installed version. Jobs using this plugin may need to be reconfigured.		

Install without restart Download now and install after restart

图36 插件管理界面

它包含四个标签:

更新:清单中列示了 Jenkins 为某些插件搜索到了可用的更新。列出的每个插件可以被选择 并应用更新。

可选安装:清单中列示了可用于安装(而不是目前已安装的)的所有插件。列出的每个插件 都可以被选择并安装。

已安装:清单中列示了已经安装的插件。

高级:允许您通过设定 HTTP 代理的方式使 Jenkins 与在线插件库建立连接。此外,还提供了一个上传设备,可以安装你在 Jenkins 以外已下载的那些插件。

由上图可知, Jenkins 缺省集成了 maven2 插件,并且一旦插件有新版本,会提示更新新版本插件。

如果想安装新的插件,可以点击 tab 分页中的可选插件。如下图:

更新	可选复作	已安裝	高级			
史裝	1			名称	版本	
rtifa	ct Uploader	rs				
	Artifacto This	ry Plugi plugin a	in allows	deploying maven artifacts and build info to Artifactory.	1.0.6	
	<u>Build Pu</u> This	blisher I plugin a	Plugin allows	records from one Hudson to be published on another Hudson.	1.8	
	CopyArc The build	s <mark>hiver P</mark> objectiv I will be	lugin ve is to copied	aggregate archived artifacts from several jobs into a shared directory. For each job, only archived artifacts of the last success I.	0.4.2	
	Deploy I This	P <mark>lugin</mark> plugin t	akes a	war/ear file and deploys that to a running remote application server at the end of a build	1.5	
	Dimensi This	ons Plug plugin i	<mark>gin</mark> ntegra	tes Hudson with <u>Dimensions</u> , the Serena SCM solution.	0.7.1	
	FTP-Pub This	lisher Pl plugin (l <mark>ugin</mark> can be	used to upload project artifacts and whole directories to an ftp server.	0.9	
	HTML Pu This	blisher plugin p	Plugin oublish	es HTML reports.	0.2.2	
	<u>java.net</u> This	upload plugin u	er Plug uses ja	in va.net tasks library to make Hudson capable of posting artifacts to java.net	1.7	
	<u>SCP plu</u> This	SCP plugin This plugin uploads build artifacts to repository sites using SCP (SSH) protocol.				
	SFEE Plu Auth	i <mark>gin</mark> enticate	users	and publish build artifacts relying on a Collabnet Source Forge Enterprise Edition (SFEE) server.	1.0.3	
	SVN Pub This	<mark>ilisher</mark> plugin a	allows	you to upload artifacts to a subversion repository. This is done via a delete/import of the items requested.	0.1	

图37 可选插件界面

从图可知, 各种 Jenkins 插件根据之前所记述的类型进行分门别类。可勾选任意想安装

的 Jenkins 插件,点击 Install without restart 按钮进行安装。安装后,所有插件以 hpi 作为后缀名放置在 plugins 文件夹下。如果是高级用户还可以自行开发插件方便具体项目使用。 注意:安装完成后需要重启 Jenkins 部署的容器。这样才能使用新装的插件。

4.2 Jenkins 插件安装示例

Jenkins 运行自动部署 war 包到 servlet 容器内,要实现这个功能必须安装一个插件。

3드-191 수-개		K +			
द्र≔ह ↓ Antifact	- Tuloadare	版平			
	Appaloes Plugin	1.4.0			
	Artifactory Plugin This plugin allows deploying Maven 2, Maven 3, Ivy and Gradle artifacts and build info to the Artifactory artifacts manager.				
	Backlog Plugin This plugin integrates <u>Backlog</u> to Jenkins.	1.9			
	Build Publisher Plugin This plugin allows records from one Jenkins to be published on another Jenkins.	1.17			
	Confluence Publisher Plugin This plugin allows you to publish build artifacts as attachments to an Atlassian Confluence wiki page.				
	<u>Crittercism dSYM Plugin</u> A Jenkins CI plugin for uploading dSYM files to Crittercism.	1.1			
V	Deploy Plugin This plugin takes a war/ear file and deploys that to a running remote application server at the end of a build	1.9			
	Deploy WebSphere Plugin This plugin is an extension of the <u>Deploy Plugin</u> . It takes a war/ear file and deploys that to a running remote WebSphere Application Server at the end of a build.	1.0			
	Dimensions Plugin This plugin integrates Hudson with Dimensions, the Serena CM solution. *{_}Please note - the maintainer for this plugin is no longer myself (TPayne), but is being changed. Until it has been updated, please forward any issues to Paul Caruana (pcaruana@serena.com)*	0.8.1			
	FTP-Publisher Plugin This plugin can be used to upload project artifacts and whole directories to an ftp server.	1.2			
Inst	3. 从证Install Without restart Download now and install after restart				

图38 安装插件界面

安装插件/升级

准备

- Checking internet connectivity
- Checking update center connectivity
- Success

Deploy Plugin 🛛 🕋 等待

Go back to the top page (you can start using the installed plugins right away)

🔶 🔲 当安装成功并且没有运行中的任务时重启Jenkins

图39 插件安装界面

好了,到此 Deploy Plugin 插件安装完成!