



# 持续交付与Jenkins应用

主讲：老钱

霍格沃兹测试学院



## Jenkins 上节内容回顾

- ✓ 持续集成与持续交付的基本概念
- ✓ Jenkins服务的搭建方法：
  - 直接启动方式
  - Tomcat 部署方式
  - Docker部署方式
- ✓ 创建基础Jenkins任务的方法



## Jenkins 课程前驱知识点

- ✓ Java程序开发
- ✓ git 源码管理工具使用
- ✓ github 的配置与使用
- ✓ 已完成github的ssh证书配置
- ✓ Maven 的配置与使用



# Jenkins 课程前驱知识点

- ✓ Jenkins的安装 + plugin安装
- ✓ Jenkins的基础使用方法
- ✓ 邮箱配置的一些基本概念：smtp, 用户名, 密码, 地址, 端口
- ✓ Spring Web应用程序基础
- ✓ 部署Web应用
- ✓ Docker 容器、镜像的操作方法
- ✓ Docker 镜像的构建方法



# Agenda

- 1 Jenkins 管理、配置运行节点
- 2 Jenkins Email配置与任务报告发送
- 3 Jenkins 多任务关联运行
- 4 Jenkins 运行单元测试项目
- 5 Jenkins 自动化部署Web应用
- 6 Jenkins 自动化部署Docker应用



分享课题：持续交付与Jenkins应用

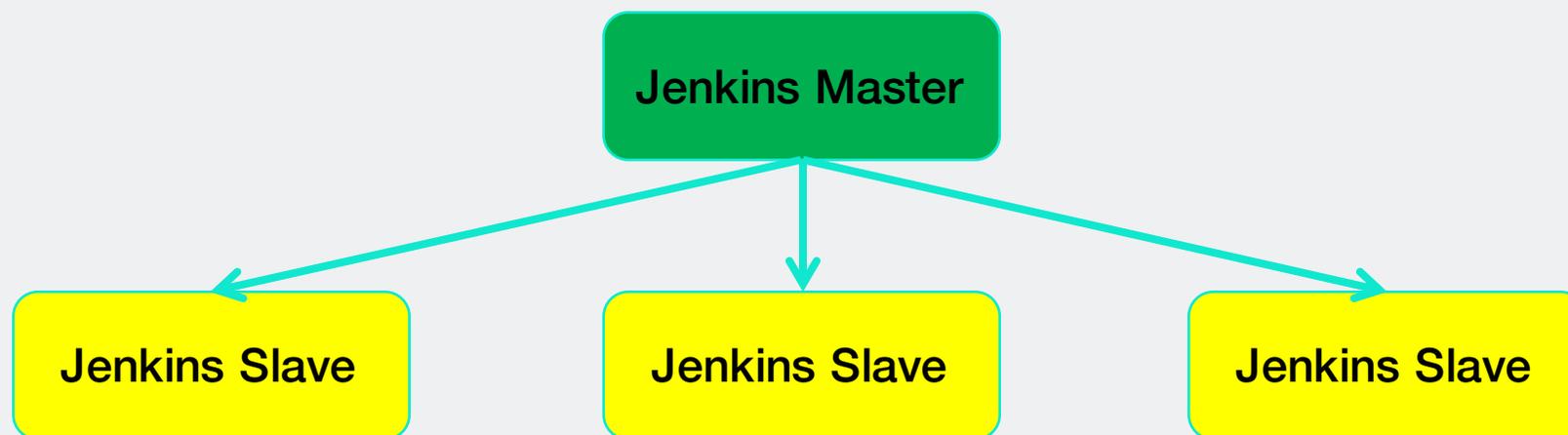
# Jenkins 管理、配置运行节点

# 01



# Jenkins 运行节点管理

- ✓ Jenkins的任务可以分布在不同的节点上运行
- ✓ 节点上需要配置Java运行时环境 , `Java_Version > 1.5`
- ✓ 节点支持 Windows, Linux, Mac
- ✓ Jenkins运行的主机在逻辑上是 master 节点





# Jenkins 运行节点管理

- ✓ 新的节点要与Jenkins Server保持连接畅通
- ✓ 支持远程运行 比如ssh , jnlp
- ✓ 远程用户名密码登录， 或者 公钥私钥认证
- ✓ 配置位置 “系统管理” -> “管理节点”

The screenshot shows the Jenkins configuration page for a new node named "TestEnv". The configuration includes the following fields:

- 名称: TestEnv
- 描述: 192.168. [redacted]
- 并发构建数: 1
- 远程工作目录: /root/jenkins
- 标签: [empty]
- 用法: 尽可能的使用这个节点
- 启动方式: Launch slave agents via SSH
- 主机: 192.168. [redacted]
- Credentials: root [redacted] Add
- Host Key Verification Strategy: Known hosts file Verification Strategy

A "保存" (Save) button is located at the bottom left of the form.



# Jenkins 运行节点管理

- ✓ 完成配置之后在节点列表刷新状态，确定Jenkins已与节点“握手”成功

[允许自动刷新](#)

S	名称 ↓	架构	时钟差异	剩余磁盘空间	剩余交换空间	剩余临时空间	响应时间	
	<a href="#">master</a>	Linux (amd64)	In sync	36.77 GB	3.73 GB	44.63 GB	0ms	
	<a href="#">TestEnv</a>	Linux (amd64)	In sync	24.48 GB	1.93 GB	24.48 GB	75ms	
	获取到的数据	3分49秒	3分49秒	3分49秒	3分49秒	3分49秒	3分49秒	

握手成功

[刷新状态](#)



## 节点管理 – git 配置

- ✓ Git 基础配置

```
git config --global user.name "your_account"
```

```
git config --global user.email "youremail@example.com"
```

- ✓ ssh-key 证书配置

```
ssh-keygen -t rsa -C "youremail@example.com"
```

- ✓ github认证通过，往github里面配public key

- ✓ 关于know\_hosts, 尝试与github 握个手哦

```
ssh git@github.com
```

- ✓ 以上配置的目的：配通 github



## 节点管理 – 配置Maven

- ✓ 建议装 Java1.8
- ✓ maven安装最新版就可以
- ✓ 如果通过copy/paste安装 还需要配置 profile 单独加载环境变量
- ✓ 如果默认的maven依赖包下载太慢可以考虑使用国内的maven库提速

```
<!-- 阿里云仓库 -->
<mirror>
  <id>nexus-aliyun</id>
  <mirrorOf>*</mirrorOf>
  <name>Nexus aliyun</name>
  <url>http://maven.aliyun.com/nexus/content/groups/public</url>
</mirror>
```



# 节点管理 - 配置Maven

```
E:\apache-maven-3.3.9\conf\settings.xml - Notepad++ [Administrator]
文件(F) 编辑(E) 搜索(S) 视图(V) 编码(N) 语言(L) 设置(T) 工具(O) 宏(M) 运行(R) 插件(P) 窗口(W) ?
new 1 settings.xml
157 |-->
158 |<mirrors>
159 |<!-- mirror
160 | Specifies a repository mirror site to use instead of a given repository. The
161 | repository that
162 | this mirror serves has an ID that matches the mirrorOf element of this mirror. IDs
163 | are used
164 | for inheritance and direct lookup purposes, and must be unique across the set of
165 | mirrors.
166 |
167 |<mirror>
168 |   <id>mirrorId</id>
169 |   <mirrorOf>repositoryId</mirrorOf>
170 |   <name>Human Readable Name for this Mirror.</name>
171 |   <url>http://my.repository.com/repo/path</url>
172 |</mirror>
173 |-->
174 |<!-- profiles
175 | This is a list of profiles which can be activated in a variety of ways, and which
176 | can modify
```

Ln : 176 Col : 14 Sel : 0 | 0 Unix (LF) UTF-8 INS



## Demo ，演示Jenkins节点配置过程，包含：

- 配置节点基础内容
- 配置 git 保持与 github 之间的畅通
- 配置 java
- 配置 maven



## 本章小结

- ✓ Jenkins节点配置方法
- ✓ 节点上配通github
- ✓ 节点上配置Java与Maven
- ✓ 提速 – 使用Maven国内库



## 练习时间

- ✓ 配置一个你自己的Jenkins节点，并配通github
- ✓ ssh-keygen 生成自己的ssh-key
- ✓ 可以使用master主机当做slave 节点



# Agenda

- 1 Jenkins 管理、配置运行节点
- 2 Jenkins Email配置与任务报告发送
- 3 Jenkins 多任务关联运行
- 4 Jenkins 运行单元测试项目
- 5 Jenkins 自动化部署Web应用
- 6 Jenkins 自动化部署Docker应用



分享课题：持续交付与Jenkins应用

# Email配置与任务报告发送

# 02



# Jenkins 配置Email

- ✓ 用户Jenkins构建任务之后发送email通知
- ✓ 配置邮件发送规则
- ✓ 配置邮件模板
- ✓ 需要下载插件



# Jenkins 配置Email

- ✓ 需要下载的Email插件名称：Email Extension, Email Extension Template
- ✓ 这两个插件的作用是帮助用户方便的设置格式化邮件

The screenshot shows the Jenkins plugin marketplace interface. A red box highlights two plugins: **Email Extension** and **Email Extension Template**. The **Email Extension** plugin description states: "This plugin is a replacement for Jenkins's email publisher. It allows to configure every aspect of email not sent, who should receive it and what the email says". The **Email Extension Template** plugin description states: "This plugin allows administrators to create global templates for the Extended Email Publisher". Other visible plugins include Device Watcher, Dimensions, Dingding[钉钉], Extreme Notification, and Google Calendar. At the bottom, there are buttons for "直接安装", "下载待重启后安装", and "立即获取", along with a timestamp "9分22秒之前获取了更新信息".



# Jenkins 配置Email

- ✓ 第一步，配置发送邮件的基础信息
  - 在“系统设置” -> “邮件通知”
  - smtp服务器
  - 用户名
  - 密码
  - smtp端口
  - 字符集
  - .....

# Jenkins 配置Email



## 邮件通知

SMTP服务器

smtp.qq.com

用户默认邮件后缀

@qq.com

使用SMTP认证

用户名

[redacted]@qq.com

密码

.....

使用SSL协议

SMTP端口

465

Reply-To Address

11547200@qq.com

字符集

UTF-8

通过发送测试邮件测试配置

Test e-mail recipient

[redacted]@qq.com

Email was successfully sent

Test configuration



# Jenkins 配置Email

- ✓ System Admin e-mail address 配置与邮箱用户名相同

## Jenkins Location

Jenkins URL

http://127.0.0.1:8081/

System Admin e-mail address

jenkins@qq.com



# Jenkins 配置Email

✓ 配置好之后可以“通过发送测试邮件测试配置”是否正确

通过发送测试邮件测试配置

Test e-mail recipient

Email was successfully sent 

[Test configuration](#)

< 收件箱  

**Test email #6**

 [详情](#)

This is test email #6 sent from Jenkins



## Email 模板配置

- ✓ 第二步，Jenkins可以根据你配置的邮件模板格式发送结果邮件
- ✓ 通过Jenkins的参数定制自己的Email模板，常用的参数key值如下：
  - \$BUILD\_STATUS – 构建结果
  - \$PROJECT\_NAME – 构建脚本名称
  - \$BUILD\_NUMBER – 构建脚本编号
  - \$JOB\_DESCRIPTION – 构建项目描述
  - \$CAUSE – 脚本启动原因
  - \$BUILD\_URL – 脚本构建详情URL地址



# Email 模板配置

- ✓ 邮件模板需要在 “系统设置” -> “Extended E-mail Notificatio” 中设置

Default Subject	← 模板标题	Jenkins运行提示: \$BUILD_STATUS   脚本名称: \$PROJECT_NAME   运行编号: \$BUILD_NUMBER
Maximum Attachment Size		
Default Content	← 模板内容	<pre>&lt;hr/&gt; (本邮件是程序自动下发, 请勿回复! ) &lt;br/&gt;&lt;hr/&gt; 项目名称: \$PROJECT_NAME&lt;br/&gt;&lt;br/&gt; 项目描述: \$JOB_DESCRIPTION&lt;br/&gt;&lt;br/&gt; 运行编号: \$BUILD_NUMBER&lt;br/&gt;&lt;br/&gt; 运行结果: \$BUILD_STATUS&lt;br/&gt;&lt;br/&gt; 触发原因: \${CAUSE}&lt;br/&gt;&lt;br/&gt; 构建日志地址: &lt;a href="\${BUILD_URL}console"&gt;\${BUILD_URL}console&lt;/a&gt;&lt;br/&gt;&lt;br/&gt; 构建地址: &lt;a href="\$BUILD_URL"&gt;\$BUILD_URL&lt;/a&gt;&lt;br/&gt;&lt;br/&gt; 详情: \${JELLY_SCRIPT,template="html"}&lt;br/&gt; &lt;hr/&gt;</pre>



# Jenkins 配置Email

- ✓ 当Jenkins的运行结果命中邮件的发送条件时，系统就会自动的发送提醒邮件

**Jenkins运行提示: Fixed | 脚本名称: wiki  
登录验证 | 运行编号: 54559**

[详情](#)

---

(本邮件是程序自动下发, 请勿回复!)

---

项目名称: wiki登录验证

项目描述: 验证wiki登录

运行编号: 54559

运行结果: Fixed

触发原因: Started by timer



# 邮件触发条件

- ✓ 触发条件：发送邮件的前提
- ✓ 邮件发送的太多等于没发，邮箱会被Jenkins邮件淹没

Default Triggers

- Aborted
- Always
- Before Build
- Failure - 1st
- Failure - 2nd
- Failure - Any
- Failure - Still
- Failure - X
- Failure -> Unstable (Test Failures)
- Fixed
- Not Built
- Script - After Build
- Script - Before Build
- Status Changed
- Success
- Test Improvement
- Test Regression
- Unstable (Test Failures)
- Unstable (Test Failures) - 1st
- Unstable (Test Failures) - Still
- Unstable (Test Failures)/Failure -> Success



# Jenkins 配置Email

**Demo , 演示Jenkins的Email 配置**



# Jenkins 参考Email配置模板

## ✓ Subject

Jenkins运行提示：\$BUILD\_STATUS | 脚本名称：\$PROJECT\_NAME | 运行编号：\$BUILD\_NUMBER

## ✓ Content

<hr/> ( 本邮件是程序自动下发，请勿回复！ ) <br/> <hr/>

项目名称：\$PROJECT\_NAME<br/> <br/>

项目描述：\$JOB\_DESCRIPTION<br/> <br/>

运行编号：\$BUILD\_NUMBER<br/> <br/>

运行结果：\$BUILD\_STATUS<br/> <br/>

触发原因：\${CAUSE}<br/> <br/>

构建日志地址：<a href= "\${BUILD\_URL}console" >\${BUILD\_URL}console</a> <br/> <br/>

构建地址：<a href= "\$BUILD\_URL" >\$BUILD\_URL</a> <br/> <br/>

详情：\${JELLY\_SCRIPT,template="html"}<br/> <hr/>



## 练习时间

- ✓ 配置你Jenkins中的email



# Agenda

- 1 Jenkins 管理、配置运行节点
- 2 Jenkins Email配置与任务报告发送
- 3 Jenkins 多任务关联运行
- 4 Jenkins 运行单元测试项目
- 5 Jenkins 自动化部署Web应用
- 6 Jenkins 自动化部署Docker应用



分享课题：持续交付与Jenkins应用

# Jenkins 多任务关联运行

# 03



## Jenkins 多任务关联运行

- ✓ 任务启动的触发条件：其他任务的运行结果
- ✓ 前驱任务成功的条件下被触发
- ✓ 前驱任务失败的条件下被触发
- ✓ 前驱任务不稳定的条件下被触发
- ✓ 适用场景：有先后次序关系的任务，比如 部署环境任务 -> 验收测试任务



# Jenkins 多任务关联运行

- ✓ 举例1：Demo 前驱任务稳定成功的条件下被触发

## Build Triggers

Trigger builds remotely (e.g., from scripts)

Build after other projects are built

Projects to watch

Trigger only if build is stable

Trigger even if the build is unstable

Trigger even if the build fails

Build periodically

GitHub hook trigger for GITScm polling

Poll SCM



# Jenkins 多任务关联运行

- ✓ 举例2：Demo 前驱任务失败的情况下被触发

## Build Triggers

- Trigger builds remotely (e.g., from scripts)
- Build after other projects are built

Projects to watch

- Trigger only if build is stable
- Trigger even if the build is unstable
- Trigger even if the build fails
- Build periodically
- GitHub hook trigger for GITScm polling
- Poll SCM

## Build

### Execute shell

Command 

```
echo "Test Fail"
exit -1
```

See [the list of available environment variables](#)



# Jenkins 多任务关联运行

- ✓ 小插曲：如何设定Jenkins任务失败 ???
  - ❖ shell 退出状态值非0, Jenkins 即会将任务状态标记为失败



# Jenkins 多任务关联运行

- ✓ 举例3：Demo 前驱任务不稳定的条件下也被触发

## Build Triggers

Trigger builds remotely (e.g., from scripts)

Build after other projects are built

Projects to watch

Trigger only if build is stable

Trigger even if the build is unstable

Trigger even if the build fails

Build periodically

GitHub hook trigger for GITScm polling

Poll SCM

## Console Output

```
Started by user admin
Running as SYSTEM
Building remotely on slave\_jenkins (group1) in workspace /Users/jj
[FirstRun] $ /bin/sh -xe /var/folders/n5/9rb1mdhs49bdj8jbp17qr4_h0
+ echo unstable
unstable
+ exit 0
[Text Finder] Scanning console output...
+ echo unstable
[Text Finder] Finished looking for pattern 'unstable' in the console
Build step 'Text Finder' changed build result to UNSTABLE
Triggering a new build of SecondRun
Finished: UNSTABLE
```



# Jenkins 多任务关联运行

- ✓ 小插曲：如何故意“设置” Jenkins任务结果不稳定 ???
  - ❖ 安装 Text Finder插件
  - ❖ 任务最终结果为成功，输出字样中含“unstable”
  - ❖ 在Post Action中设置规则：如果在console中检测到“unstable”，则标记任务状态为unstable

### Build

**Execute shell**

Command `echo "unstable"`  
`exit 0`

See [the list of available environment variables](#)

Add build step ▾

### Post-build Actions

**Text Finder**

Files

Also search the console output

Regular expression

Succeed if found

Unstable if found

Not Built if found

Add post-build action ▾



## Jenkins 多任务关联运行 - 本章小结

- ✓ 多任务关联运行适用场景 – 有先后执行次序关系的任务
- ✓ 触发条件 ....
- ✓ 处理Jenkins 任务结果状态的几个小技巧



# Agenda

- 1 Jenkins 管理、配置运行节点
- 2 Jenkins Email配置与任务报告发送
- 3 Jenkins 多任务关联运行
- 4 Jenkins 运行单元测试项目
- 5 Jenkins 自动化部署Web应用
- 6 Jenkins 自动化部署Docker应用



分享课题：持续交付与Jenkins应用

# Jenkins 运行单元测试项目

# 04



## 单元测试项目简介

- ✓ 保存在github上：<https://github.com/princeqjzh/iTest>
- ✓ 基于JUnit的接口测试程序，使用的程序语言 java
- ✓ 需要使用Maven进行构建与执行
- ✓ 调试程序的时候建议使用Intellij Idea



## 单元测试项目简介

# Demo , 演示单元自动化测试程序



# 单元测试运行环境准备

- ✓ Linux
- ✓ git
- ✓ ssh 证书 -> github
- ✓ java
- ✓ maven
- ✓ IntelliJ IDEA



# Jenkins 单元测试任务创建

- ✓ 创建Job
- ✓ 基础配置
- ✓ 添加源码
- ✓ 设定触发条件
- ✓ 添加构建代码
- ✓ 添加结果报告



# Jenkins 单元测试任务运行

## Project iTest



工作区



最新修改记录



Disk Usage

Job	301 KB
All builds	301 KB
Locked builds	-
All workspaces	57 KB
Slave workspaces	57 KB
Non-slave workspaces	-

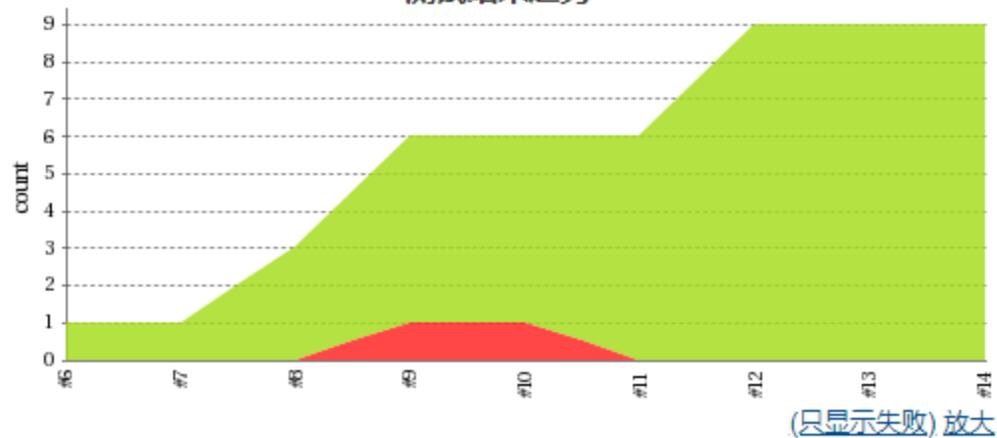


最新测试结果 (没有失败)

添加说明

禁用项目

测试结果趋势



(只显示失败) 放大



# Jenkins 单元测试任务演示

## Demo , 演示创建单元测试任务



## 本章小结

- ✓ JUnit自动化测试项目
- ✓ 自动化测试的运行方法
- ✓ Jenkins任务需要配置的内容
- ✓ 测试报告能够告诉我们的信息
- ✓ 常见的问题



# Agenda

- 1 Jenkins 管理、配置运行节点
- 2 Jenkins Email配置与任务报告发送
- 3 Jenkins 多任务关联运行
- 4 Jenkins 运行单元测试项目
- 5 Jenkins 自动化部署Web应用
- 6 Jenkins 自动化部署Docker应用



分享课题：持续交付与Jenkins应用

# Jenkins 自动化部署Web应用

# 05



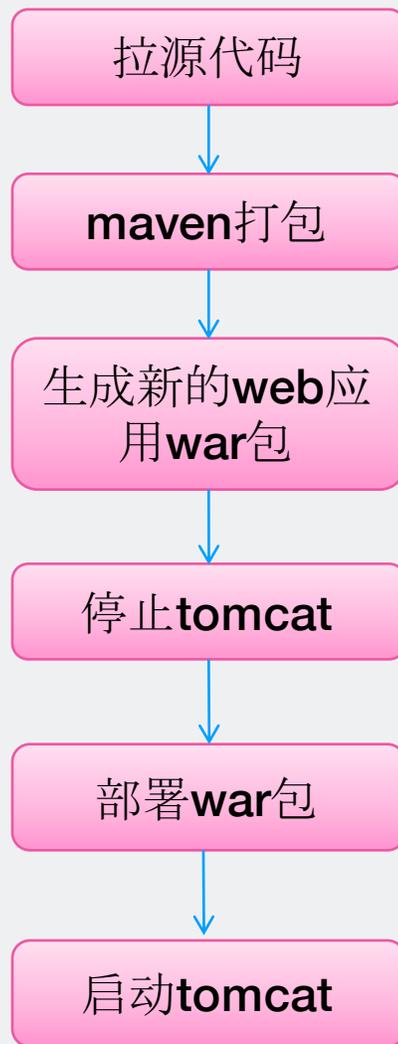
## Web 应用程序简介

- ✓ 保存在github上：<https://github.com/princeqjzh/iWeb>
- ✓ 基于Spring的Web应用程序
- ✓ maven打包生成war文件
- ✓ 运行项目中的shell脚本，部署web应用war包到 Tomcat
- ✓ 重启tomcat
- ✓ 确定Web应用启动正确



# 演示iWeb应用程序

## Demo , 演示iWeb程序





## 详解部署 Shell 代码

# Demo , 部署流程与shell 代码



## iWeb 部署的Jenkins任务

- ✓ Job基础信息配置
- ✓ 获取源码
- ✓ 配置shell命令
- ✓ 运行调试
- ✓ 别忘了 `BUILD_ID=DONTKILLME`



# iWeb 部署的Jenkins任务

## Demo , 调试Jenkins job



## 本章小结

- ✓ 关于Web应用，打包步骤
- ✓ Web应用的部署流程
- ✓ 部署的Shell代码编写
- ✓ Tomcat base 发布 = 编译 + 打包 + 部署 + 重新启动Tomcat



# Agenda

- 1 Jenkins 管理、配置运行节点
- 2 Jenkins Email配置与任务报告发送
- 3 Jenkins 多任务关联运行
- 4 Jenkins 运行单元测试项目
- 5 Jenkins 自动化部署Web应用
- 6 Jenkins 自动化部署Docker应用



分享课题：持续交付与Jenkins应用

# Jenkins 自动化部署Docker应用

# 06

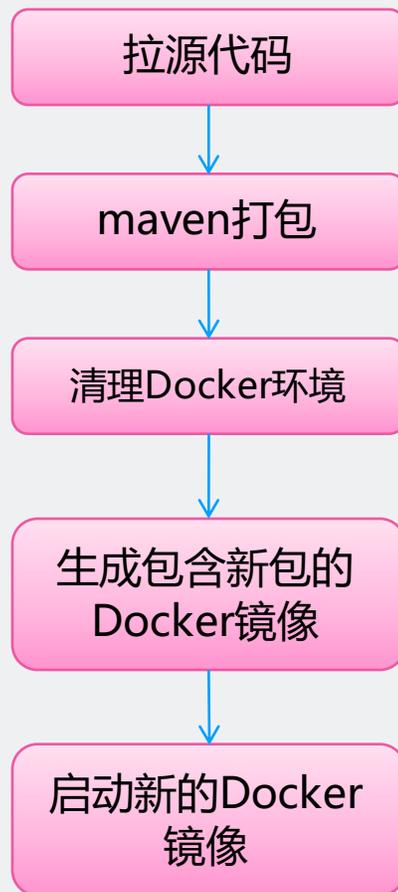


# Docker Web 应用简介

- ✓ Web应用 – 还是上一章所使用的 iWeb
- ✓ 部署的不同点 – 将iWeb应用程序包发布到docker中然后运行
- ✓ 与之前部署操作的相同点
  - ❖ 编译、打包的过程相同
  - ❖ Jenkins任务创建过程雷同
- ✓ 与之前部署操作的不同点
  - ❖ 使用Dockerfile 生成部署应用的docker image
  - ❖ 需要通过命令来控制docker container的运行状态



# Docker Web 应用部署流程





# Docker Web 应用部署

**Demo , docker 部署代码讲解 与 创建Jenkins 任务过程演示**



## 本章小结

- ✓ 关键之一 控制docker
- ✓ 利用Dockerfile 生成部署web应用的docker image
- ✓ 通过shell 代码控制整体过程
- ✓ Docker 发布 = 编译 + 打包 + 清理docker环境 + 生成image + 启动docker container



# Agenda

- 1 Jenkins 管理、配置运行节点
- 2 Jenkins Email配置与任务报告发送
- 3 Jenkins 多任务关联运行
- 4 Jenkins 运行单元测试项目
- 5 Jenkins 自动化部署Web应用
- 6 Jenkins 自动化部署Docker应用



- ✓ Jenkins 运行节点管理、配置
- ✓ Jenkins Email 配置、模板与触发条件
- ✓ 多任务关联
- ✓ Jenkins运行单元测试
- ✓ Jenkins部署Web应用与Docker应用
- ✓ Jenkins 对于实际工作中的一些帮助



谢谢

THANKS