多玩YY事业部软件包规范

V0.1

广州华多网络有限公司

2012年1月

目录

[一、简介 3](#_Toc315864948)

[适用范围 3](#_Toc315864949)

[适用对象 3](#_Toc315864950)

[为什么要使用(软件)包 3](#_Toc315864951)

[使用(软件)包的好处 3](#_Toc315864952)

[二、术语介绍 4](#_Toc315864953)

[包 4](#_Toc315864954)

[实例 4](#_Toc315864955)

[全新安装 4](#_Toc315864956)

[正式升级 4](#_Toc315864957)

[测试升级 5](#_Toc315864958)

[发布模式的对比 5](#_Toc315864959)

[三、包的生命周期管理 5](#_Toc315864960)

[包的生命周期 5](#_Toc315864961)

[包的创建流程 6](#_Toc315864962)

[包的上线流程 6](#_Toc315864963)

[包的下线流程 7](#_Toc315864964)

[三、包管理规范细节 7](#_Toc315864965)

[基础信息规范 7](#_Toc315864966)

[目录规范 8](#_Toc315864967)

[日志规范 9](#_Toc315864968)

[版本号规范 10](#_Toc315864969)

[启停规范 10](#_Toc315864970)

[端口规范 11](#_Toc315864971)

[命名规范 11](#_Toc315864972)

[配置规范 12](#_Toc315864973)

[部署规范 12](#_Toc315864974)

[四、包监控规范 12](#_Toc315864975)

[五、模块间调用规范(暂不讨论) 13](#_Toc315864976)

[六、发布流程(暂不讨论) 13](#_Toc315864977)

# 一、简介

## 适用范围

该规范适用于广州华多网络科技有限公司 YY 事业部

## 适用对象

开发、运维人员

## 为什么要使用(软件)包

* 环境标准化
* 提高IDC操作效率
* 为后续 D/O 分离打下基础

## 使用(软件)包的好处

**环境标准化**

* 所有程序均按照相同的目录结构部署
* 所有程序遵循相同的管理方式
* 所有程序遵循相同的数据上报规范
* IDC 所有程序信息的统一管理

**提高IDC操作效率**

通过未来的包管理系统, 实现自动打包,减少手工 scp 操作

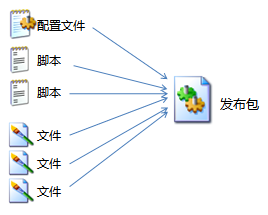
配合包管理系统,实现一键化的安装、升级、卸载、起停等操作

软件包自带进程存活监控、MD5校验等脚本,节省开发、运维编写脚本的时间

# 二、术语介绍

## 包

包是一组目标执行代码文件、脚本和配置文件，按照一定的规则（打包规范）组成的可以自行安装和自行部署的组合



## 实例

一个包安装到机器上就成为一个实例

## 全新安装

* 以软件包 A 为例， 如果目标机器上不存在软件包 A 的任何版本，则称为”全新安装”
* 适用于首次安装某个包。全新安装不限制版本，不一定需要从最低版本开始安装

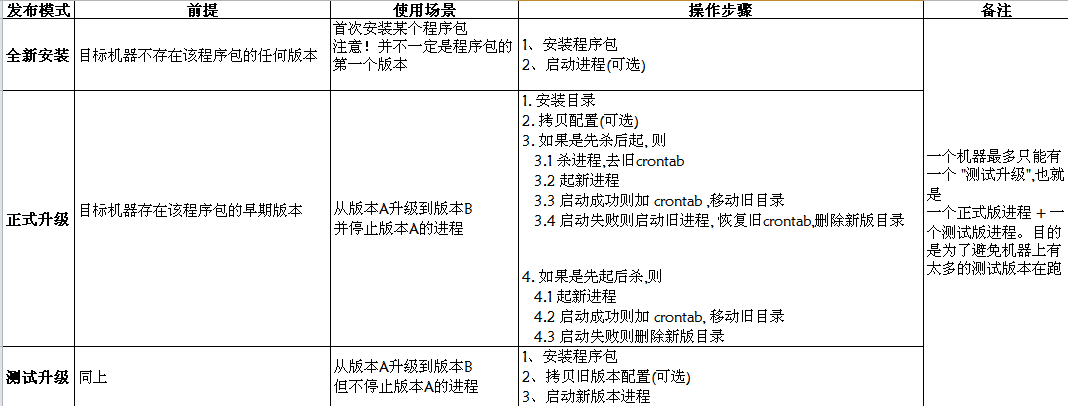
## 正式升级

* 以软件包 A 为例，如果目标机器安装了软件包A 的v1 版本， 现在升级到 V2 版本
* 正式升级会停止老版本的进程
* 适用于正式版本的上线

## 测试升级

* 以软件包 A 为例，如果目标机器安装了软件包A 的v1 版本， 现在升级到 V2 版本
* 测试升级**不会**停止老版本的进程
* 适用于测试版本的上线

## 发布模式的对比



# 三、包的生命周期管理

## 包的生命周期



## 包的创建流程



**注 : CMC 就是配置管理中心，也就是后续要建设的”配置管理系统”**

## 包的上线流程



**注 : 第2步(leader 审核) 与第三步 “测试环境部署” 在现阶段暂时没有**

## 包的下线流程



# 三、包管理规范细节

## 基础信息规范

|  |  |  |
| --- | --- | --- |
| **字段** | **举例** | **备注** |
| **名 称** | slave\_session\_media | 1、不包含版本、平台、操作系统信息  2、一律小写、字母、下划线组成 |
| **版本号** | 4.4.3 | 1、这里指的是该软件包本身的版本,不是里面所包含的二进制版 |
| **进 程**  **名 称** | slave\_session\_media.64\_V4.4.3 | 1. 该字段对应软件包所包含的需要被管理的二进制程序文件名,必须严格与包的内容对应。 2. 该字段存在的作用是为了标识软件包中,那些进程希望通过自动化脚本管理,例如自动启动、停止、重启等。 3. 如果一个软件包包含了多个二进制文件,但仅有个别二进制需要自动管理,可以只填写对应的名称,无需全部写上 4. 允许填写多个进程名,彼此之间用分号隔开 |
| **运 行**  **用 户** | root | 1. 进程以什么用户身份运行 2. 尽量避免用 root 启动 3. 后续将为每个开发组分配一个业务帐号,建议每个业务组用自己的帐号启动 |
| **所 属**  **业 务** | <部门> + <小组> | 1、用于定位该软件包所属业务 |
| **操 作**  **系 统** | Ubuntu 9.04 | 1、该软件包所适用的操作系统 |
| **平 台** | 64位 | 1、软件的编译平台 |
| **描 述**  **信 息** | 媒体前端进程 | 1、描述该软件包的主要功能 |
| **Changelog** | 增加 vrouter.ko | 1、当前打包版本相对于上一版本的改动之处 |



## 目录规范



* bin 存放可执行文件
* lib 存放库文件
* conf 存放配置文件
* etc 符号链接到 conf/
* log 存放日志文件目录，符号链接到 /data/yy/log/<包>-<版本>， 安装时自创建
* data 存放程序数据, 符号链接到 /data/yy/data/<包>-<版本>，安装时自创建

**注意事项**

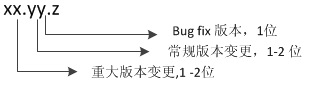
* **上述目录，包管理系统前台默认根据模板自动创建，无需人工创建**
* 若软件包有特殊类型文件则请建立文件夹存放，不得与已有文件类型混放
* 任何文件不得直接存放在软件包顶层文件夹

## 日志规范

* 所有业务日志不允许打印到操作系统的日志文件(/var/log/\*)
* **使用 log4cplus 开源组件进行日志轮转**
* **禁止使用 syslog-ng 来处理业务日志，**避免上线前忘记让运维配置 syslog-ng 从导致磁盘爆满的问题, 也减少上线的步骤，提高上线效率
* 所有日志必须打印到安装目录的 log/ 子目录下，实际存放是/data/yy/log/<包名称>-<版本>/ 下
* 日志必须分类。建议如下
  + 访问日志 ：access.log
  + 错误日志 ：error.log
* 日志内容
  + 访问日志 ：记录每个请求处理的详细日志。
  + 错误日志 ：包含进程自身错误信息(例如内存不足、磁盘只读等), 或者业务逻辑导致的错误信息(例如用户无登录权限、无访问频道权限等)
* 日志分级
  + 分成 DEBUG, INFO, WARNING , ERROR , FATAL
  + 建议正常情况下只输出 ERROR 级别
* 日志轮转
  + 建议单个日志文件的最大体积不超过300MB
  + 同一个软件包的单个版本,最多维持10份,也就是 3GB 的日志
  + 日志文件名必须是 \*.log.<n> 结尾
  + 上述功能均可以通过 log4cplus 配置实现
* 日志清理 (包管理脚本完成，无需开发实现)
  + 默认认为 log/ 下的文件都是可以自动清理的,脚本清理前不会做任何通知
  + 如果有重要数据, 请放到 data/ 目录下,避免被误删
  + 磁盘空间使用率高于80% 时将自动启动清理脚本,按时间逐个日志文件清除,直到磁盘空间可用率低于告警阀值(80%)为止。

## 版本号规范

* 软件包版本号在打包时指定, 系统不会默认填写
* 软件包版本号与所包含的二进制版本没有任何关系, 但建议保持一致
* 软件包初始版本无需从0开始,建议初始版本号等同于该软件的最新稳定版本
* 当一个软件包在现网有部署实例时, 不允许撤销该版本的软件包
* 下面是版本号数字的规定



## 启停规范

* 所有后台进程必须做成 daemon 方式, 启动后 fork 子进程,并进入后台运行。
* 将按照打包时所指定的用户启动进程
* 建议尽量不要使用 root 帐号启动和运行二进制,以避免安全问题
* 不再使用 sudo 方式启动, 而是采用由 root 执行 su - <user> -c <cmd> 的方式启动
* 启动命令默认是 ./<prog> &
* 默认启动进程数量是1个。如果需要多个, 请通过自定义启动脚本解决
* 包管理系统将提供启动前、启动后命令, 以满足实际情况需要
* 停止进程默认使用 killall <cmd> 命令

## 端口规范

* 没有必要的情况下, 不使用1024以下端口
* 被动端口 : 1025-15000
  + V4
  + 语音前端
  + 。。。
* 主动端口 : 20000-65000
* 程序应该避免使用外界常用的木马、恶意程度端口
* 严禁不同程序共用端口(例如 linkd\_d 与 imlinkd\_d )
* 程序的管理端口必须固定(否则无法实施 iptables 策略) 同时必须有 iptables 访问控制，建议也加上用户、密码控制， 避免公网用户随意访问

## 命名规范

* 所有二进制文件名只能以小写字母、下划线、数字组成。不允许其他字母、符号出现
* 二进制程序名不允许超过15个字母, 这样会造成 ps 、top 等命令显示不全, 也会给运维脚本带来困扰
* 配置文件名必须是 .conf (普通文本格式) 或者 .xml (xml格式) 结尾
* 日志文件名必须是 .log.<n> 结尾
* 数据文件名必须是 .dat 结尾
* 库文件名必须是 .so 结尾

## 配置规范

* 严禁不同程序共用同一个配置文件
* 配置文件如果包含密码等信息, 访问权限必须做控制

## 部署规范

* 同一台机器允许运行相同 server 的多个版本
  + 正式版本允许一个或多个
  + 测试版本只允许一个

# 四、包监控规范

* 程序必须提供接口以供性能查询,例如通过管理端口, 或者独立二进制的方式(例如 squidclient)
  + 上一分钟请求数: (reqs)
  + 上一分钟用户数 (users)
  + 上一分钟请求成功数、成功率 (succeedNum, succeedRate)
  + 上一分钟请求失败数、失败率 (failedNum, failedRate)
* 如果短时间内无法实现, 也可以通过日志文件的方式,单独写入 log/performance.log , 以每分钟1条的格式写入。格式是 <timestamp> <item>=<value> <item2>=<value2> <item3>=<value3> 。。。

# 五、模块间调用规范(暂不讨论)

# 六、发布流程(暂不讨论)