

SACC

2012中国系统架构师大会

SYSTEM ARCHITECT CONFERENCE CHINA 2012

架构设计 · 自动化运维 · 云计算

Linux平台软件管理系统 设计与规划

Duanjigang <duanjigang1983@gmail.com>



贺兰云天雪

内容和目标

内容:

- 1) RPM 和 YUM 介绍 (15分钟)
- 2) 管理平台设计 (25分钟)
- 3) 规范化管理 (10分钟)

目标 :

- 1) 认识 linux 平台软件管理系统
- 2) 从软件管理系统的改进体会架构设计
- 3) 企业内部软件管理系统规范概述



SACC

2012中国系统架构师大会

SYSTEM ARCHITECT CONFERENCE CHINA 2012

架构设计 · 自动化运维 · 云计算

1 RPM和YUM介绍-从程序员说起



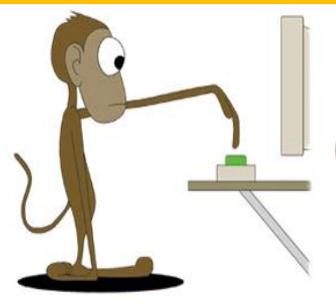
Coding is interesting

spring



Coding is all my life

summer



Boring , but I like it

autumn



Tired, striving

winter

I am happy, I am a coding monkey ...

SACC

2012中国系统架构师大会

SYSTEM ARCHITECT CONFERENCE CHINA 2012

架构设计 · 自动化运维 · 云计算

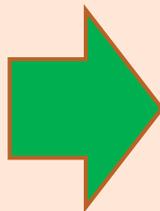
1 YUM和RPM介绍-程序员遇见运维

One day..



终于开发完鸟

哇呀呀！！这些不要问我！功能已经开发完成了！你自己用吧



亲，我该用你尼？

Coding is not all of the software engineering , There are more work after the development.....

SACC

2012中国系统架构师大会

SYSTEM ARCHITECT CONFERENCE CHINA 2012

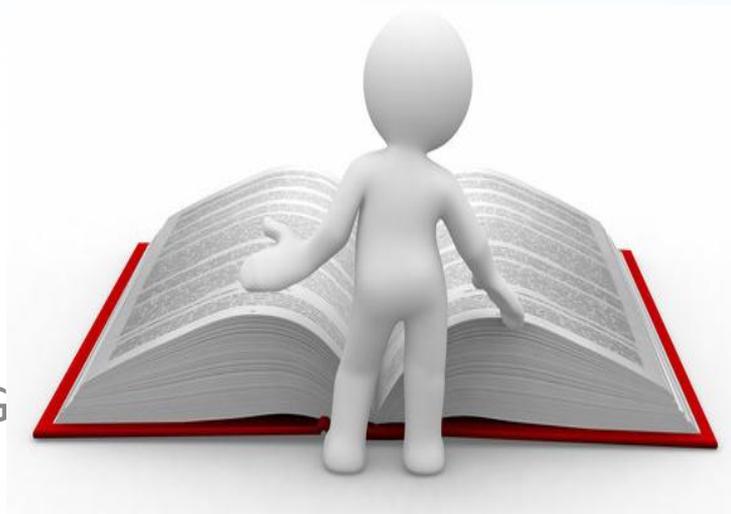
架构设计 · 自动化运维 · 云计算

1 RPM与YUM介绍-程序开发完成之后

用软件包的方式组织你的程序

常见的软件包组织方式:

- (1):目录文件方式软件包
- (2):压缩包方式 zip , tgz , rar
- (3):更高级封装,RPM, DEB, SLP, PKG

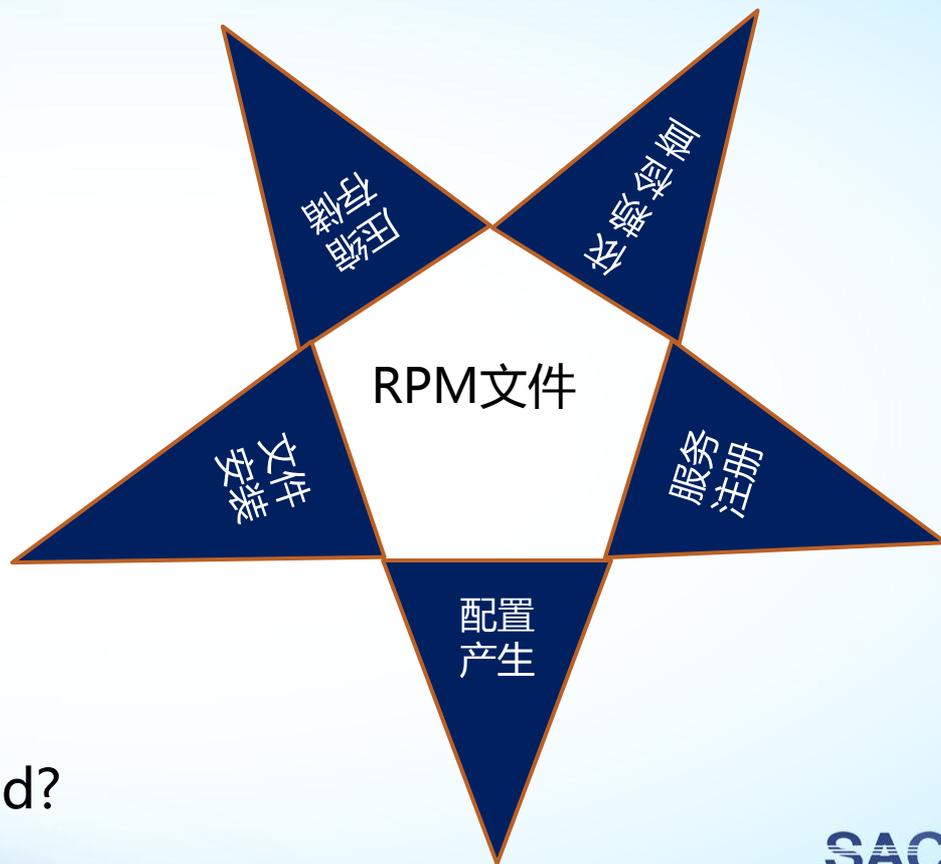


我们的故事就是从RPM这种软件包讲起的...

1 RPM与YUM介绍-软件包的特征

各种格式软件包的一般特征:

- 压缩存储数据
- 文件安装指定位置
- 配置文件产生
- 注册系统服务
- 软件依赖检查



What is an RPM ?

How is an RPM file produced?

1 RPM与YUM介绍-RPM文件的组成格式

* The lead

- 1: 存储rpm内部信息
- 2: 标识一个特定的RPM包
- 3: name不灵活-lead要被废弃

```
struct rpmlead {  
    unsigned char magic[4];  
    unsigned char major, minor;  
    short type;  
    short archnum;  
    char name[66];  
    short osnum;  
    short signature_type;  
    char reserved[16];  
};
```

Name长度改变后:

- A:新版本的RPM命令不能读取老版本的RPM信息
- B:老版本RPM命令不能安装新版本包



* The Header Structure

- 1: The Header Structure Header
存储头的起始位置和大小, 存储数据的大小
- 2: The Index Entry
每个index存储每个数据段的信息, 大小和位置, 会存在多个index
- 3: store 存储header的实际数据

The signature-RPM完整性检查等自身特征信息

The Header-包含包的所有信息, 名称, 版本, 文件列表等等

The Archive-包中所有文件, gun zip 方式存储

1 RPM与YUM介绍-YUM的出现

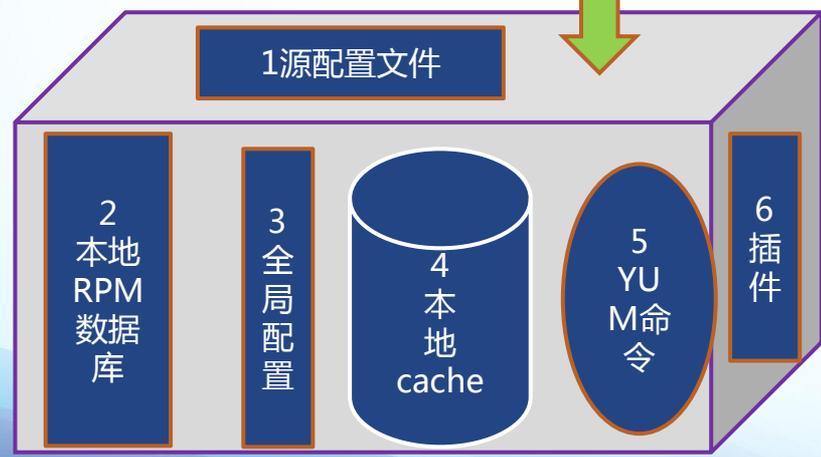
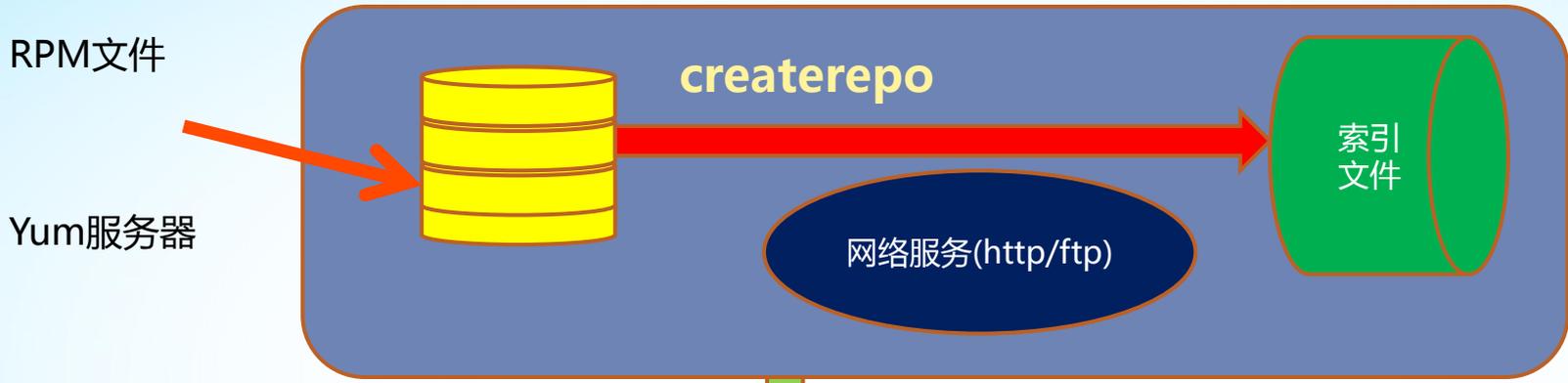
* YUM出现的原因:

- 自动处理依赖包的安装问题(权值最重)
- 软件包管理集中化，方便大规模安装
- 用户界面更简易，灵活

* YUM系统构成要素:

- 1 服务器端安装包仓库(包含仓库中的RPM文件和RPM的索引文件)
- 2 服务器端提供文件下载的网络服务(HTTP,FTP等)
- 3 客户端yum命令集合(实现server端包下载和信息查询)
- 4 客户端配置文件(服务端地址和软件仓库的配置，还有YUM的扩展模块，比如插件等)

1 RPM与YUM介绍-YUM系统构成



安装客户端

Yum list/install/search/info

- 1:/etc/yum.repos.d/*.repo
- 2:/var/lib/rpm/_db.*
- 3:/etc/yum.conf
- 4:/var/cache/yum/*
- 5:/usr/bin/yum
- 6:/usr/lib/yum-plugins/*.py和/etc/yum/pluginconf.d/*.conf

1 RPM与YUM介绍-YUM搭建和使用实例

Yumserver为192.168.1.100，一个客户端为192.168.1.101，其中yumserver机器提供http服务，web根目录为/var/www/html，两台机器都安装是32位linux系统

步骤:

- 制作好RPM包：testrpm-1.1.2-20.i386.rpm
- 把testrpm-1.1.2-20.i386.rpm拷贝到192.168.1.100的/var/www/html/yum目录下
- 进入192.168.1.100的/var/www/html/yum目录下执行:sudo createrepo --update -d -p .生成索引目录repopdata
- 在192.168.1.101上配置/etc/yum.repos.d/yumtest.repo内容为:

```
[yum]
name=yum
baseurl=http://192.168.1.100/yum
enabled=1
gpgcheck=0
```

- 在客户端执行 sudo yum clean all, yum list, yum install cmeguard进行验证

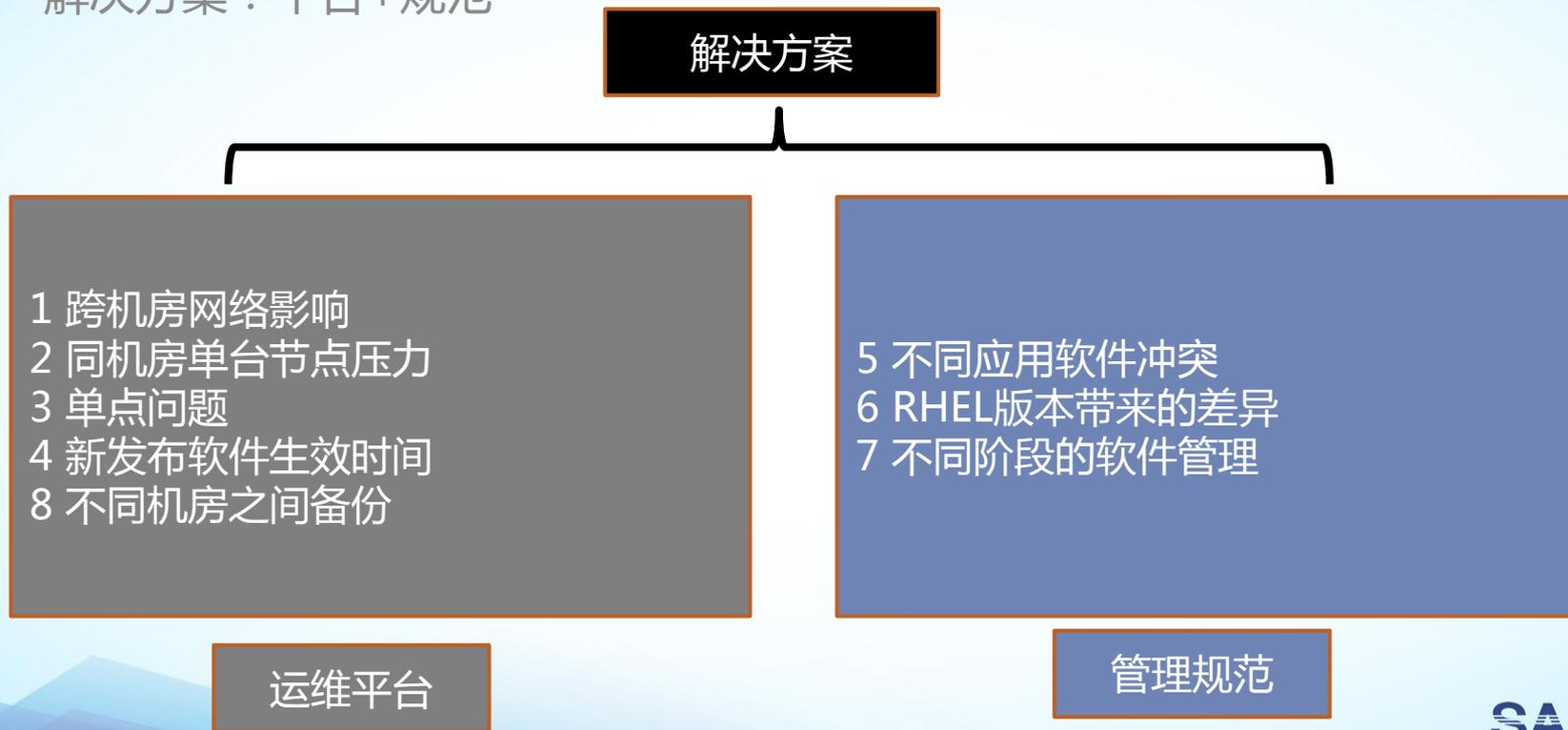
1 RPM与YUM介绍-大规模应用时遇到的问题

- 多机房应用-接近10个
- 服务器节点增加-10万台以上
- RPM 文件个数剧增-10万个左右
- OS (RHEL)多个版本-5U的5个，6U的3个，4u,3u..

- 1 跨机房安装，影响出口带宽
- 2 同机房内server节点网卡流量打满，TCP连接过高
- 3 单点问题
- 4 同一仓库下RPM过多，更新索引慢，生效时间长
- 5 应用增加，不同应用的软件包冲突
- 6 不同RHEL小版本引起软件差异
- 7 不同阶段的软件如何管理(测试版，稳定版)
- 8 不同机房之间容灾

1 RPM与YUM介绍-问题解决方案

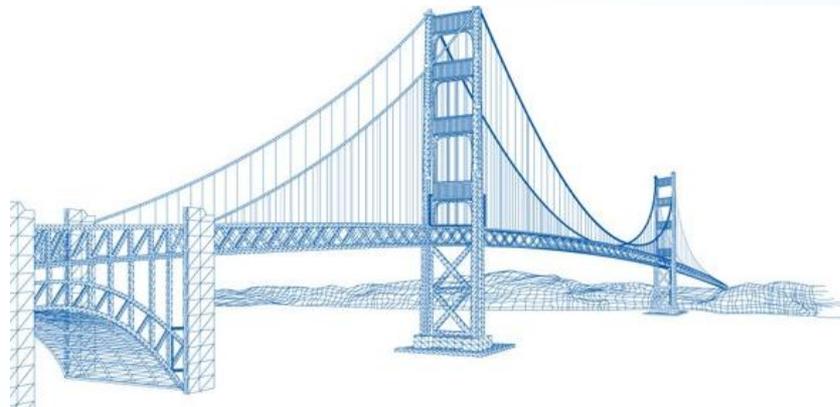
解决方案：平台+规范



2 平台设计

平台设计的原则：

- 稳定-YUM服务长期可用
- 安全-RPM文件可信
- 高效-软件包生成，发布时效性高



SACC

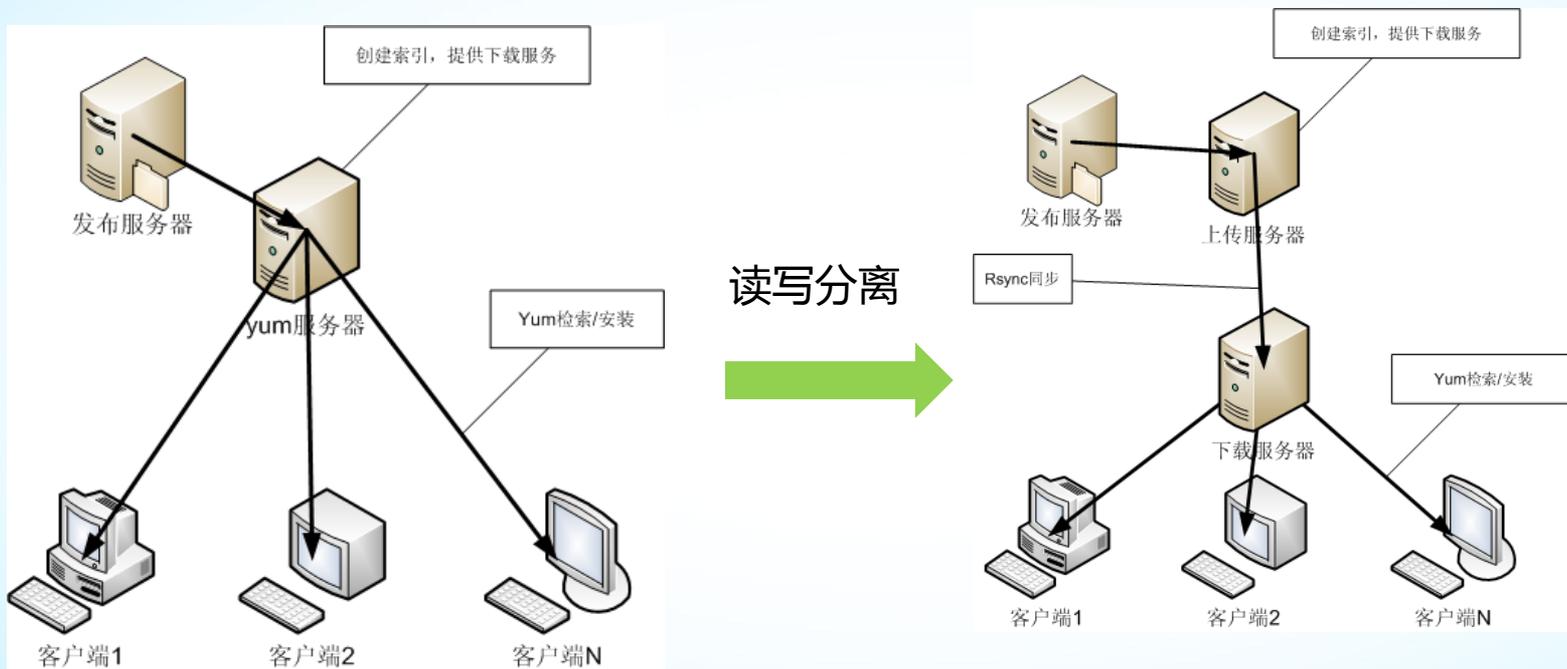
2012中国系统架构师大会

SYSTEM ARCHITECT CONFERENCE CHINA 2012

架构设计 · 自动化运维 · 云计算

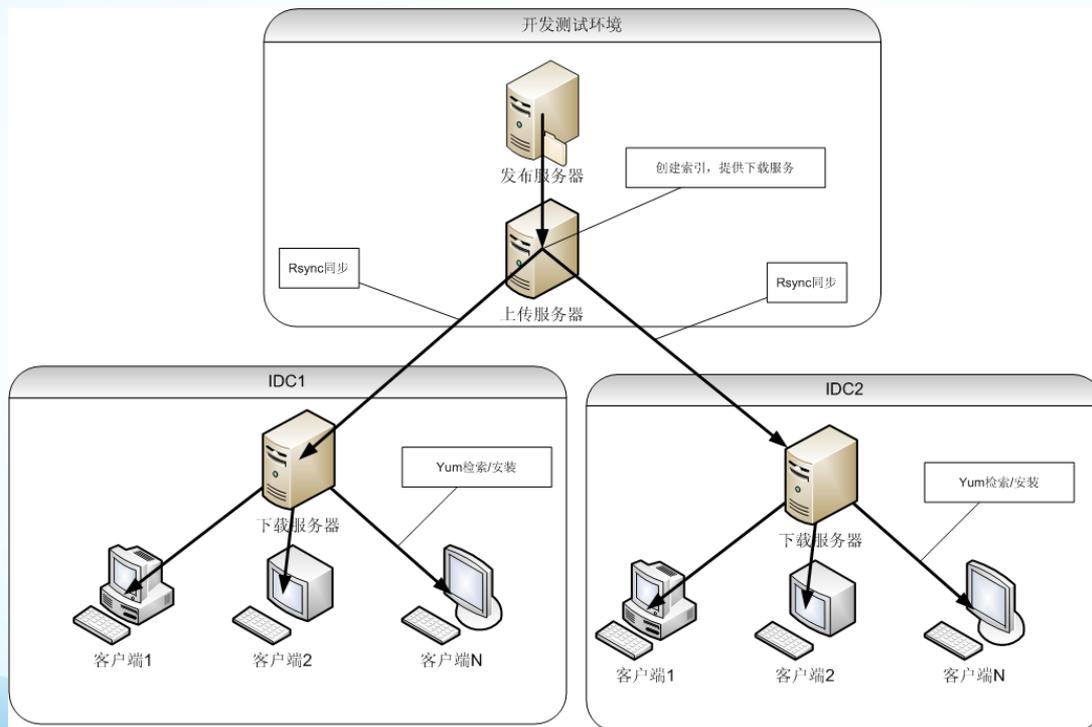
2平台设计-第一步：读写分离

读写分离:索引服务器和下载服务器独立，保障下载服务，为下载扩展做准备



2平台设计-第二步：多机房MIRROR

多机房部署mirror,减少跨机房访问



2平台设计-第三步：机房内扩展

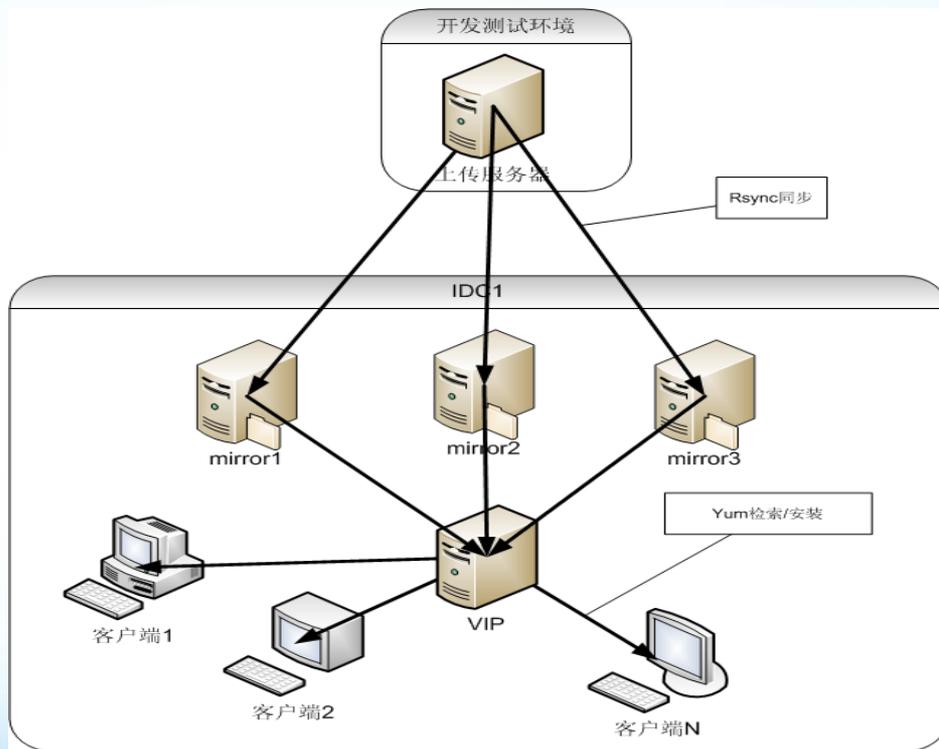
单机房内扩展:解决单点问题，单server负载过高问题

实现:

- 1 LVS绑定VIP的方式
- 2 同域名DNS轮转

问题:

- 1 多个mirror的同步一致
- 2 LVS/DNS轮转健康检查



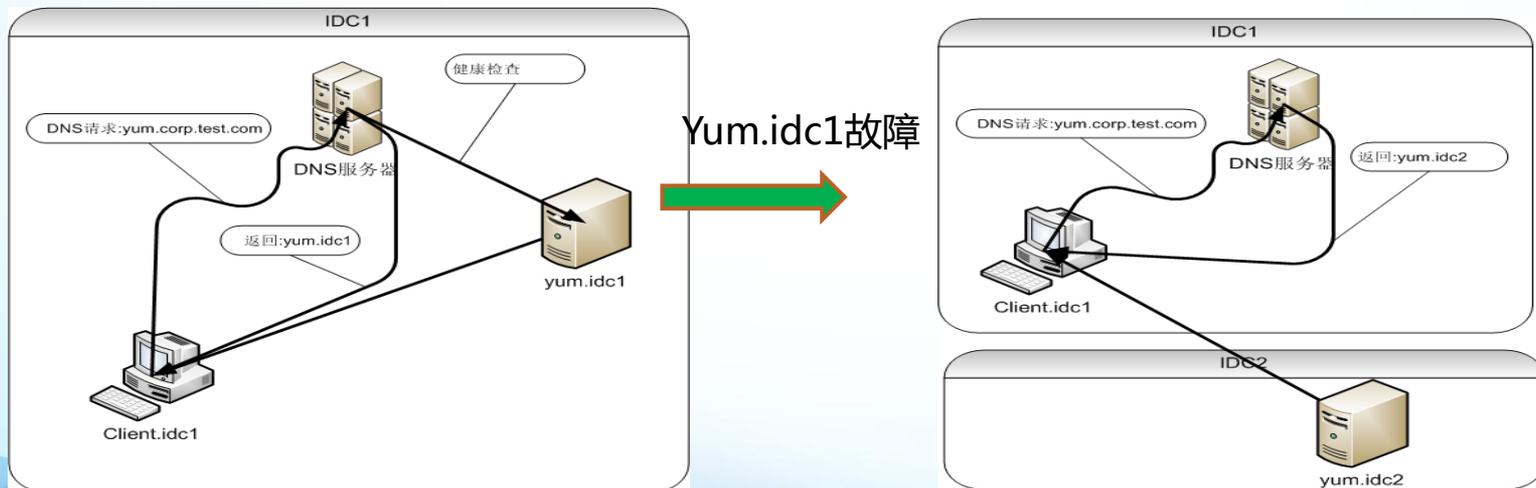
2平台设计-第四步：不同机房之间互相备份

本机房yum服务故障时，暂时访问别的机房的yum服务(网络条件)

实现:

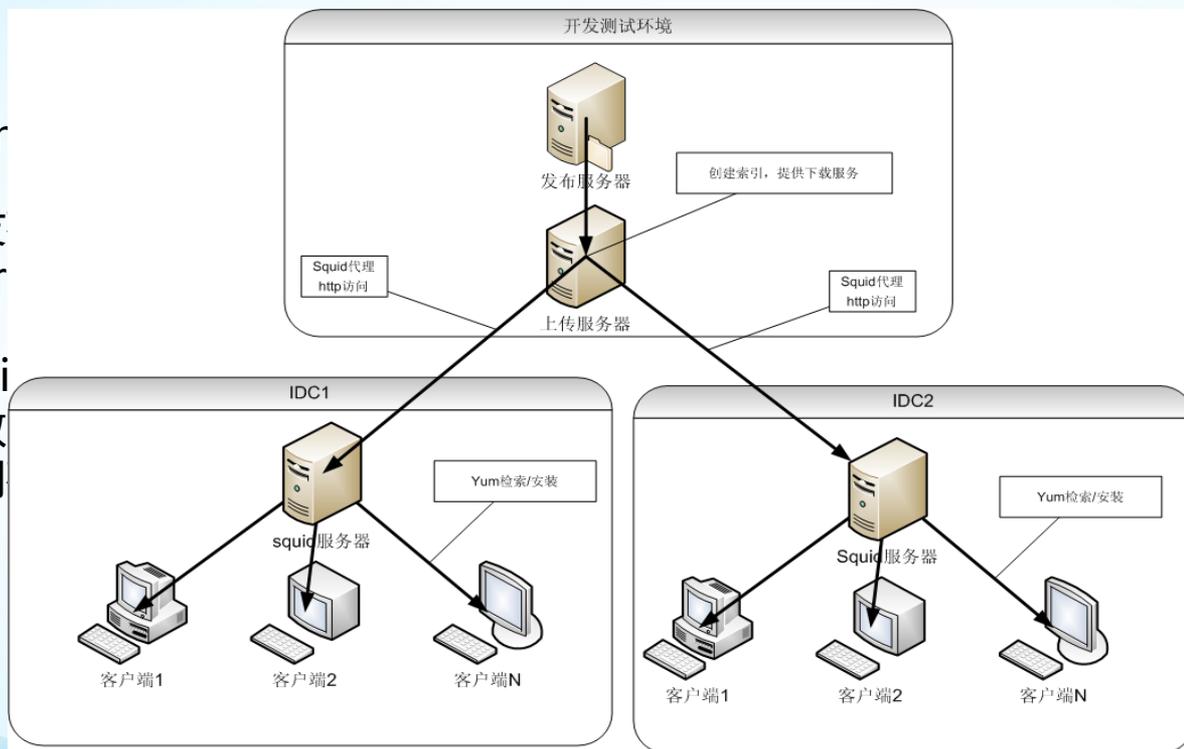
主要借助智能DNS系统:

- DNSPod
- Bind + 自实现健康检查



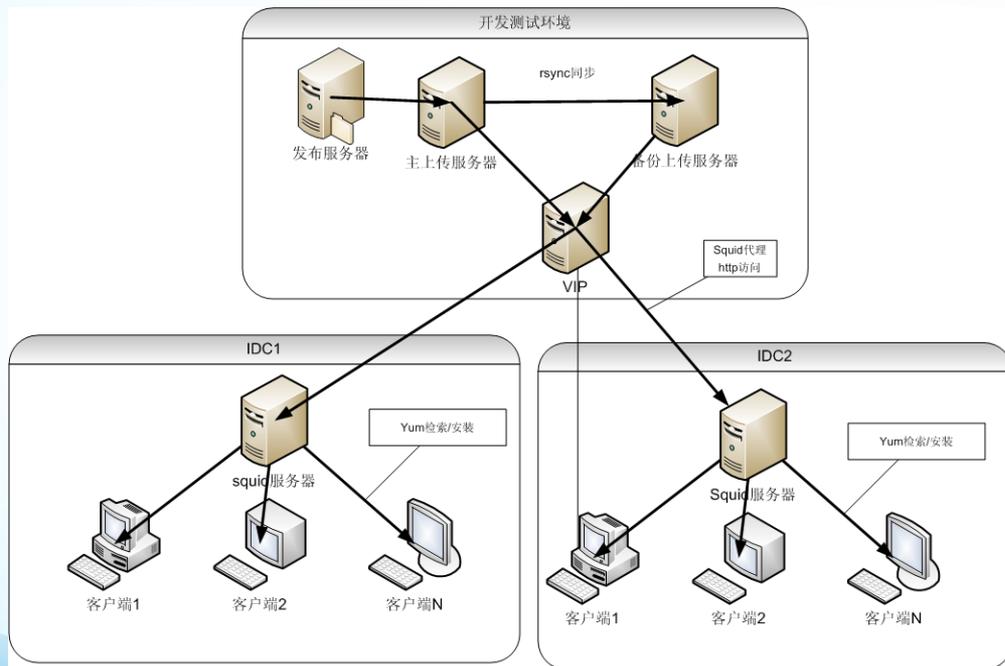
2平台设计-第五步：SQUID/TS替代RSYNC

Rsync
新发
Mirr
Squid
生效
占用



2平台设计-第六步：SQUID遗留的问题

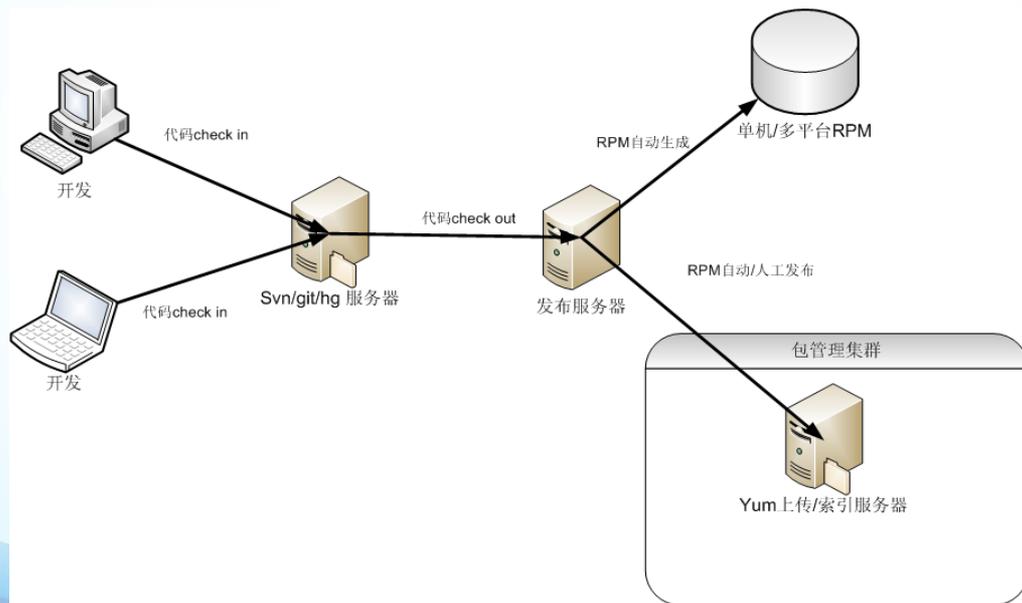
索引服务器对于squid是个单点问题，http挂掉，mirror即不可用
解决办法:索引服务器增加备份，做双机备份



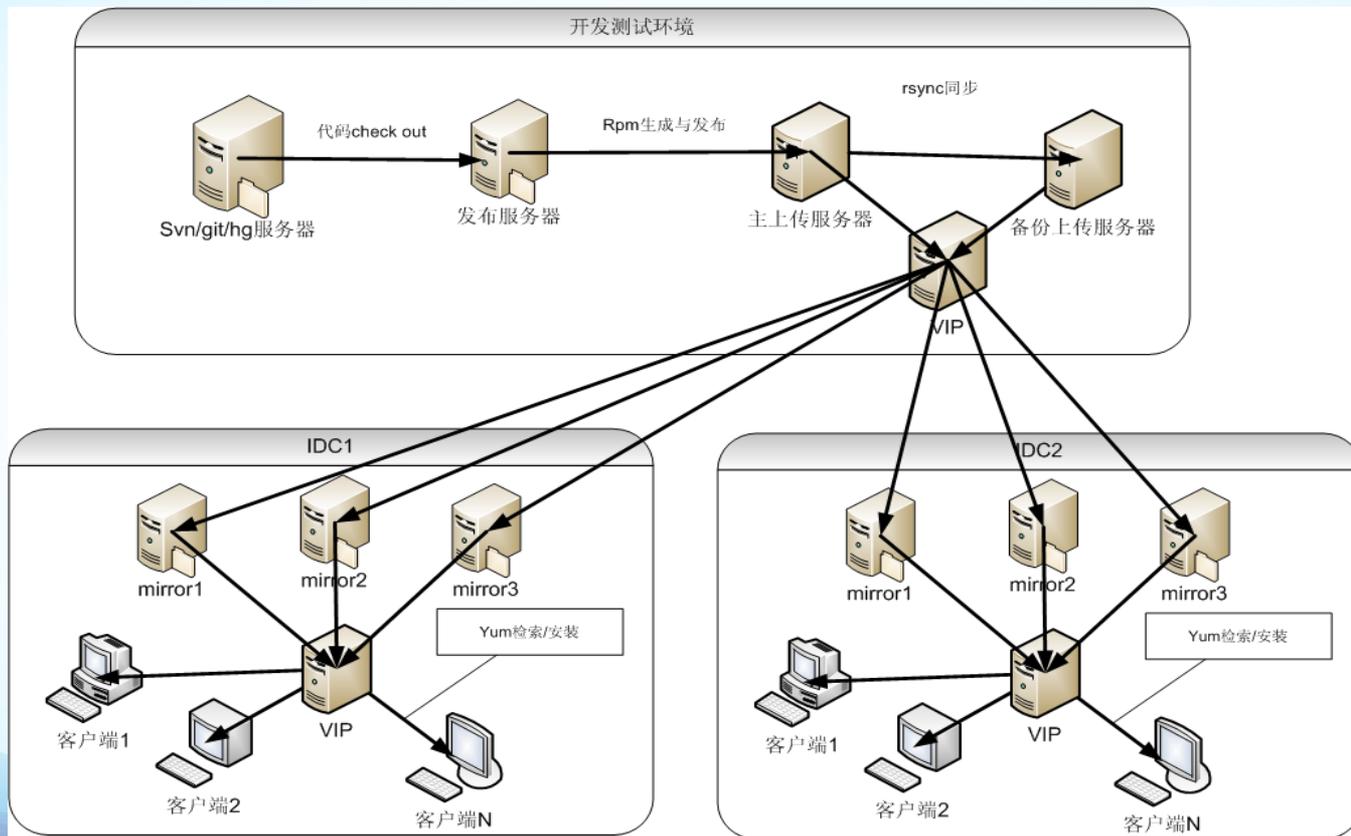
2平台设计-第七步：包制作与发布

需求:

- 1) 单机 rpm 制作工具，简单，易用(封装rpbuild)
- 2) 多平台 rpm 制作工具(i386/noarch/x86_64, rhel4/5/6)
- 3) Rpm 发布到源的工具，实现规范检查，审计等功能



2平台设计-最终结构



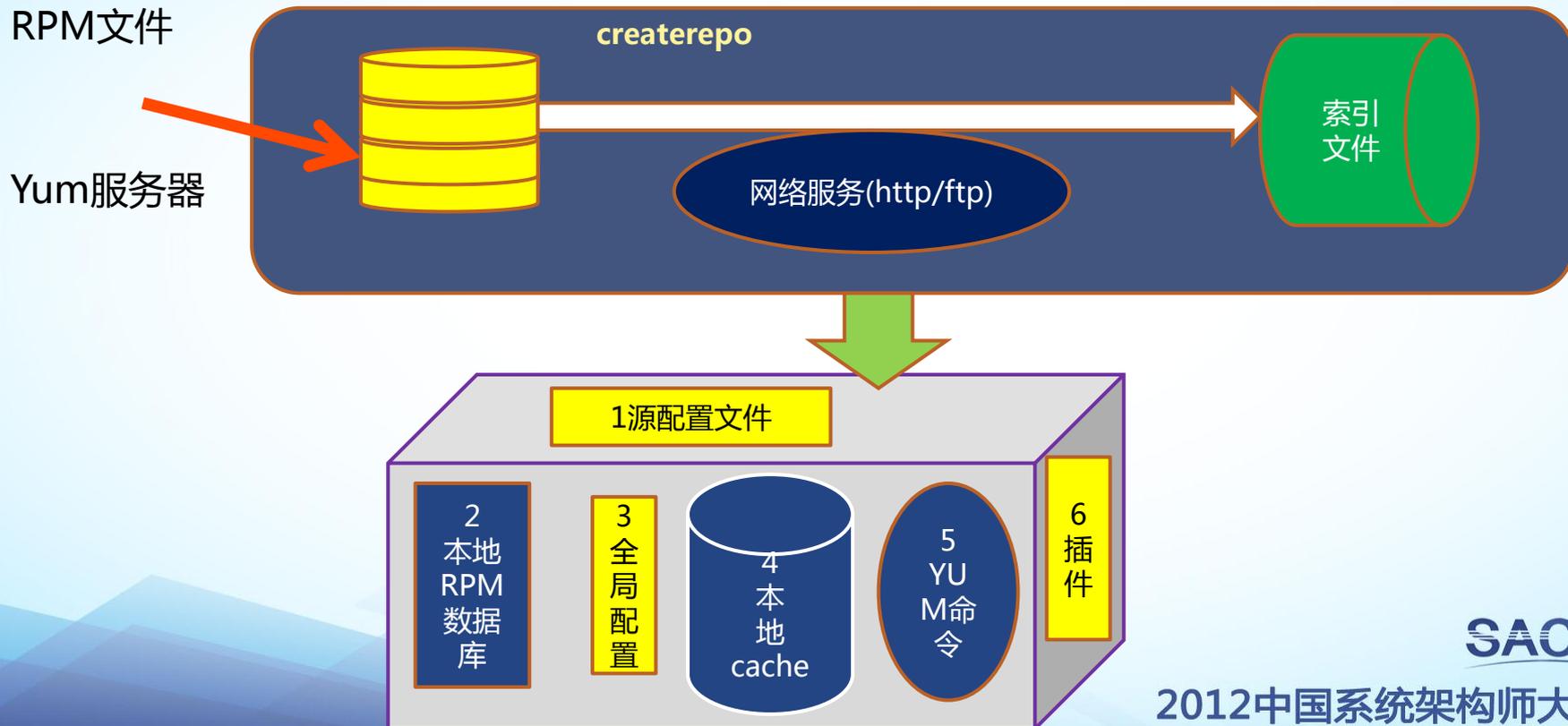
2平台设计-再回首RSYNC+HTTP与SQUID

软件选型	优势	特点	个人建议
Rsync+http	简单，稳定，自成系统	磁盘空间要求 存在同步周期， 生效慢	实时性要求不高，硬件资源足，可以选用
squid	生效时间快，磁盘空间少	依赖于上级 http，配置和 稳定性	准实时生效，硬件资源紧缺，只有虚机可用时

没有万能的结构；具体场景，具体选择！

3 规范管理-总领

对yum系统的各个组成部分进行管理(标注黄色)



3 规范管理-仓库管理

仓库进行合理划分

- 系统仓库(rhel)
- 应用仓库(app)
- 应用基础仓库(app_base)

仓库划分的好处

- 方便分拆化维护(基础软件/应用开发- , 分角色维护)
- Yum源索引更新更高效
- 方便对应用软件源做高级处理(yum插件)

SACC

2012中国系统架构师大会

SYSTEM ARCHITECT CONFERENCE CHINA 2012

架构设计 · 自动化运维 · 云计算

3 规范管理-插件管理

Yum plugin 主要作用:

- 对源/客户端配置做更高级应用
- 最常见:源的选择上进行扩展

Yum plugin 应用举例:

- 1) Priority 插件:通过优先级调整, 优先安装app源中的包
- 2) branch 插件:根据软件所处阶段, 选择从test/current/stable开始查找

开发手册:

<http://yum.baseurl.org/wiki/WritingYumPlugins>

3 规范管理-软件命名/内容管理

1 软件命名

固定前缀，方便区分来源，标识企业/部门/产品线/yum源等

比如:corp-app1-mysql,coprpr-app2-named等

2 安装路径

方便OS/磁盘分区等管理

基础包: /app_base/xxoo

应用包:/app/xxoo

3 规范管理-全局配置/源配置/插件配置管理

- /etc/yum.conf -全局配置cachedir,log,obsolete等
- /etc/yum.repos.d/*.repo-多个源，路径，GPG配置
- /etc/yum/plugin.d/*.conf -扩展插件功能支持

3 规范管理-常见问题与解决方法

- Yum域名不可访问:ping,telnet
- 索引文件不可用:wget,zcat,sqlite
- 找不到期望的包:wget,lynx --dump
- Yum/rpm阻塞住:rpmdb清除, killall yum rpm rpmq, rm -fr /va/lib/rpm/__db*
- 新包没有及时更新:yum clean all; rm -fr /var/cache/yum/*
- 多个源冲突/指定源安装: yum install -disablerepo=* --enablerepo=test
- 插件引起问题: yum install -noplugins
- Yum不可用:检查python版本, yum相关包的安装情况

参考资料

- Maximum RPM
 - <http://www.rpm.org/max-rpm/index.html>
- Packaging software with RPM, Part 1: Building and distributing packages
 - <http://www.ibm.com/developerworks/library/l-rpm1/>
- Packaging software with RPM, Part 2: Upgrading and uninstalling software
 - <http://www.ibm.com/developerworks/library/l-rpm2/>
- WritingYumPlugins
 - <http://yum.baseurl.org/wiki/WritingYumPlugins>
- RPM Guide
 - http://docs.fedoraproject.org/en-US/Fedora_Draft_Documentation/0.1/html/RPM_Guide/
- RPM specs file syntax
 - [http://wiki.mandriva.com/en/Policies/RpmSpecProposal#Requires.2C_Obsoletes.2C_Provides.2C_Conflicts.2C_et](http://wiki.mandriva.com/en/Policies/RpmSpecProposal#Requires.2C_Obsoletes.2C_Provides.2C_Conflicts.2C_et_c.)
c.

提问和解答？

Thanks

SACC

2012中国系统架构师大会

SYSTEM ARCHITECT CONFERENCE CHINA 2012

架构设计 · 自动化运维 · 云计算