Introduction To PUPPET

Plan

- Puppet: Just Simple
  - features | components | installation|usage
- Puppet: Never Leave
  - living example file and service management
- Maybe Puppet, Maybe Other, Maybe None

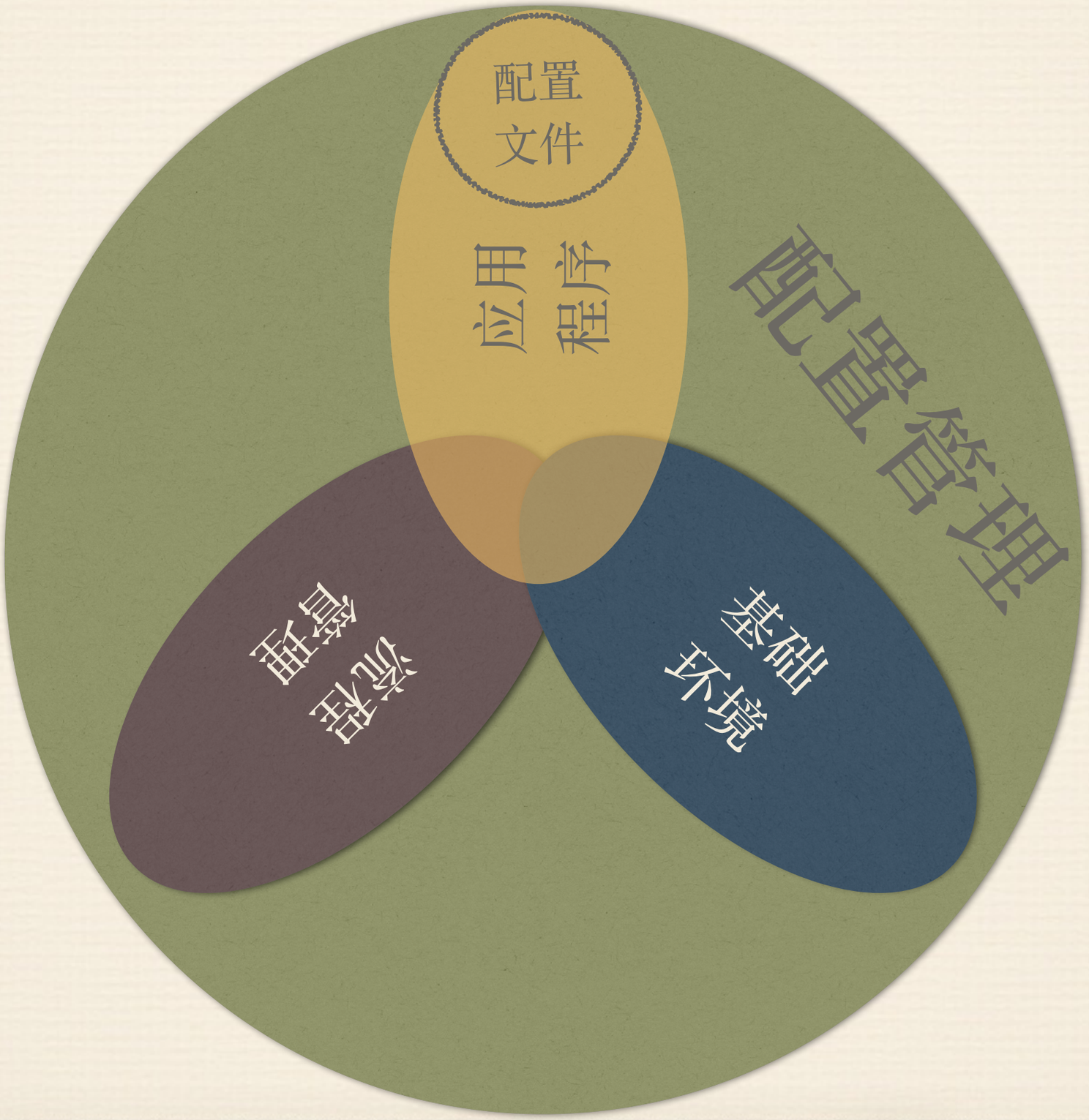
Just Simple

- Tool Chain
- Birth Of Puppet
- Features
- Components
- WorkShop
- Ecosphere

配置管理的变迁

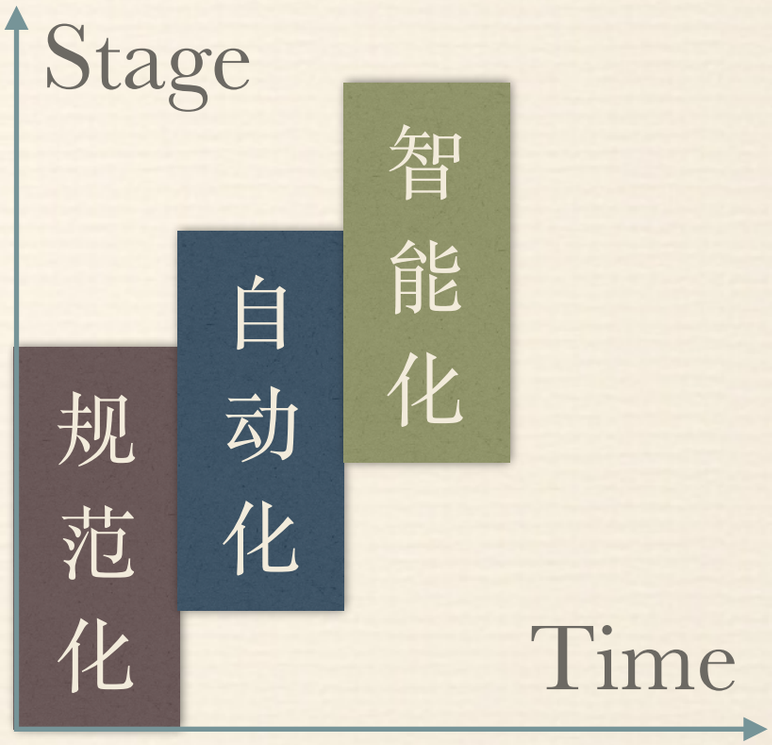
- 配置文件—版本控制|配置模版
- 应用程序—组件管理|依赖管理
- 流程管理—构建|测试|部署
- 基础环境—用户|组|服务|包

图解包含关系



配置管理的变迁

- 命令行
- 脚本
- 工具
- 平台
- Dream - - - True Cloud



工具链介绍

系统安装	配置管理	命令编排	监控报警[1]
Kickstart(Linux) Jumpstart (Solaris) YaST(SUSE) Cobbler	CFEngine Chef Puppet SaltStack AnsibleWorks	Fabric Capistrano Func MCollective SaltStack AnsibleWorks	Cacti Ganglia Nagios Graphite Zabbix

[1]http://en.wikipedia.org/wiki/Comparison_of_network_monitoring_systems

Three Ways Comparisons

Method	Puppet MCollective	SaltStack	AnsibleWorks
Native Lang	Ruby	Python	Python
Distributed	C/S	C/S	Agentless
Desc Lang	DSL(Puppet)	Yaml	Yaml
Template	ERB	Jinja2/Mako	Jinja2
CMD Via	AMQP	ZeroMQ	SSH
CMD Speed	Faster	Faster	Fast
CMD Sec	Muti Chooices	AES	SSH
Community	Mature	Active	Active
Use Case	OpenStack Docker		
User	Google/Intel	Apple/Hulu	
Complexity	Difficult	Medium	Simple

[1]http://missingm.co/2013/06/ansible-and-salt-a-detailed-comparison/

Father Of PUPPET

- Why Puppet Birth
  - CFengine && ISconf
- Why Puppet in Ruby
  - Perl Vs Python Vs Ruby

[1]http://junqili.com/puppet/introduction-to-puppet/

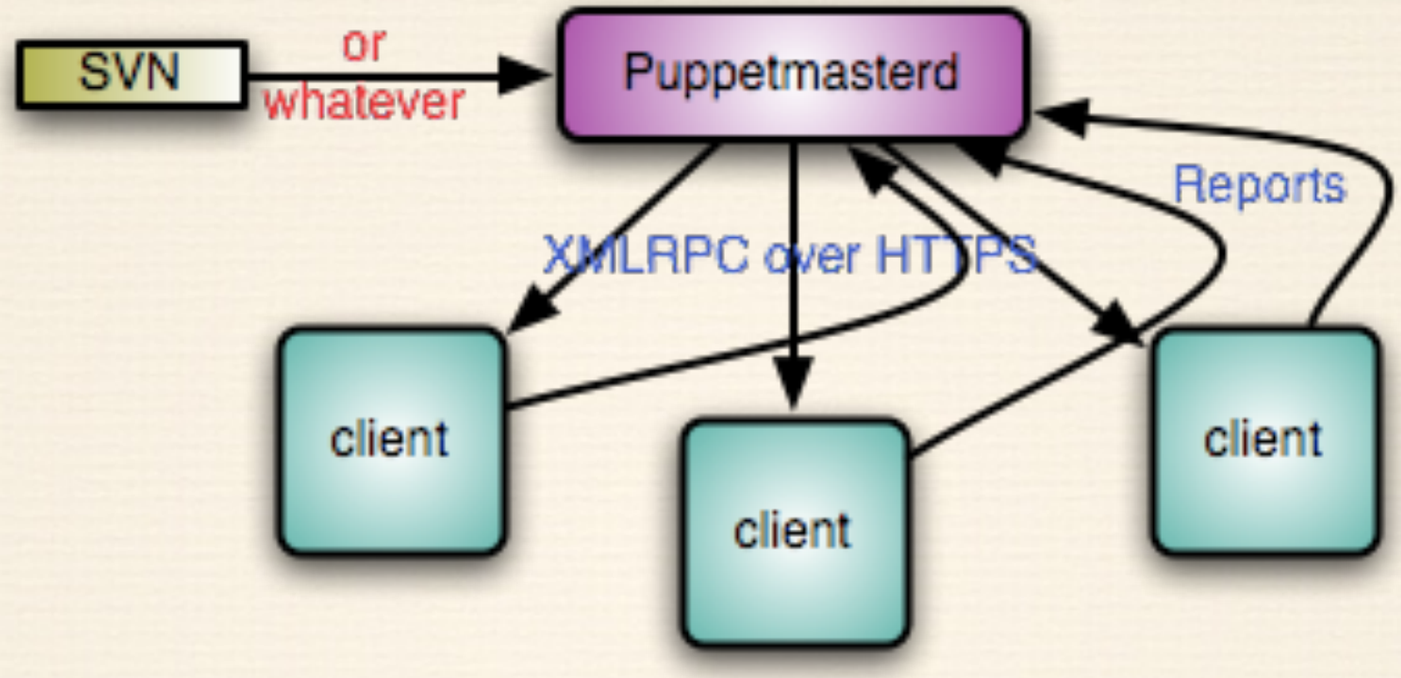
What Is Puppet

+ Puppet是一个配置并维护你的计算机的工具

+ 使用它简单的配置语言，你向Puppet解释你所希望的机器配置参数，然后它将根据需要更改配置来匹配你的要求

+ 如果你的配置有所变化，比如包更新，Puppet将自动更新你的机器来匹配。如果它们已经按照要求配置，那么Puppet就会什么也不做。

C/S Software/Application



- Server/Agent mode
- Server Less mode

Why Puppet

- 强大的框架简化大部分系统管理工作
- 使用Puppet语言的代码有很好复用性

Plugin-in System/Framework

- 增加新类型（资源）
- 为已有类型增加后端
- 插件化系统和开放API

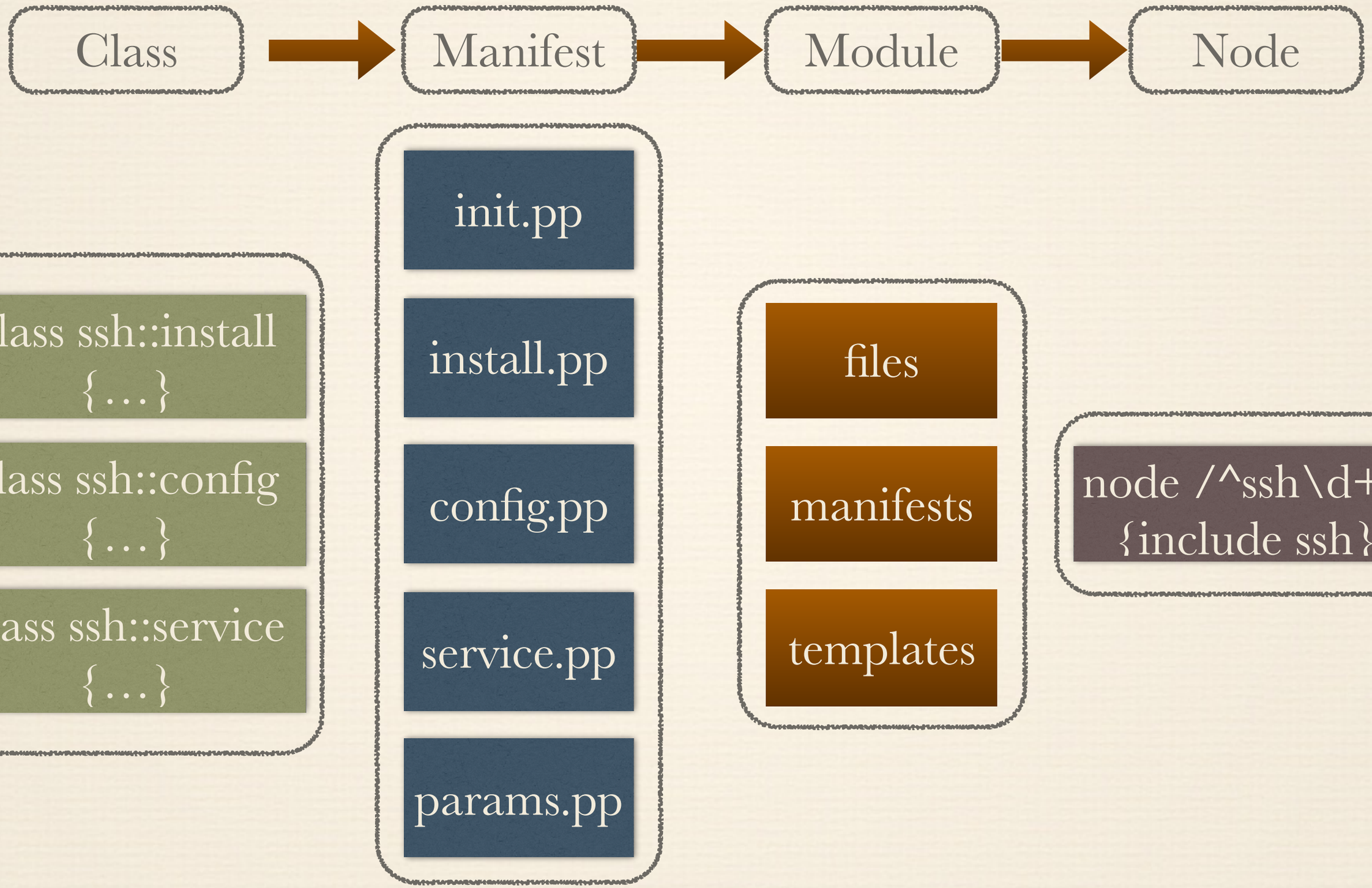
DSL Language

- Use Ruby
- Declarative
- You Specify Configuration, Puppet handles implementation
  - Use detailed specification
  - Need things like dependencies

Code Snippet

```
class ssh {
    package { ssh: ensure => installed }
    file { "/etc/ssh/sshd_config":
          ensure => present,
          owner => root,
          group => root,
          mode => 0755,
          source => 'puppet:///modules/ssh/sshd_config',
          require => Package["ssh"]
    }
    service { sshd:
          ensure => running,
          enable => true,
          hasrestart => true,
          hasstatus => true,
          subscribe => File["/etc/ssh/sshd_config"],
    }}
```

Organization



Example::Files

- 内容型
- 模版型

Contents

manifests/lquery.pp

```
class wap::cache{
    file{'/search/staff/daemon/cache/conf':
        ensure => directory,
        owner => staff,
        group => staff,
        mode => 0755,
    }
}
class wap::cache::files {
    file{'/search/staff/daemon/wap/conf/cache.cfg':
        require => File[‘/search/staff/daemon/cache/conf’],
        ensure => present,
        owner => staff,
        group => staff,
        mode => 0755,
        source => ‘puppet:///WapFile/cache.cfg',
    }
}
```

Templates

- **manifests/files.pp**

```
class ob::files::cache inherits
    ob::params::wap {
     file{ '/opt/pack/ob/conf/cache.cfg':
         content => template('observer/cache.cfg.erb'),
         require => File['/opt/pack/ob/conf'],
     }
}
```

- **manifests/params.pp**

```
    class ob::params::cache {
        $PreFix = 'wap'
        $ProcName = 'cache'
        $ErrNum = 48
        $LatNum = 20
        $LogLimit = 15
    }
```

Templates

templates/cache.cfg.erb

```
PREFIX="<%= @PreFix %>"
USER="odin"
BASE_DIR="/search/staff/daemon/<%= @ProcName %>"
PROC_NAME="<%= @ProcName %>"
CONF_NAME="<%= @ProcName %>.cfg"
RESTART_USER="root"
RESTART_SH="restart_<%= @ProcName %>.sh"

ERR_SAVE_NUM="<%= @ErrNum %>"
LAT_SAVE_NUM="<%= @LatNum %>"
LOG_LIMIT="<%= @LogLimit %>"
```

Innovations

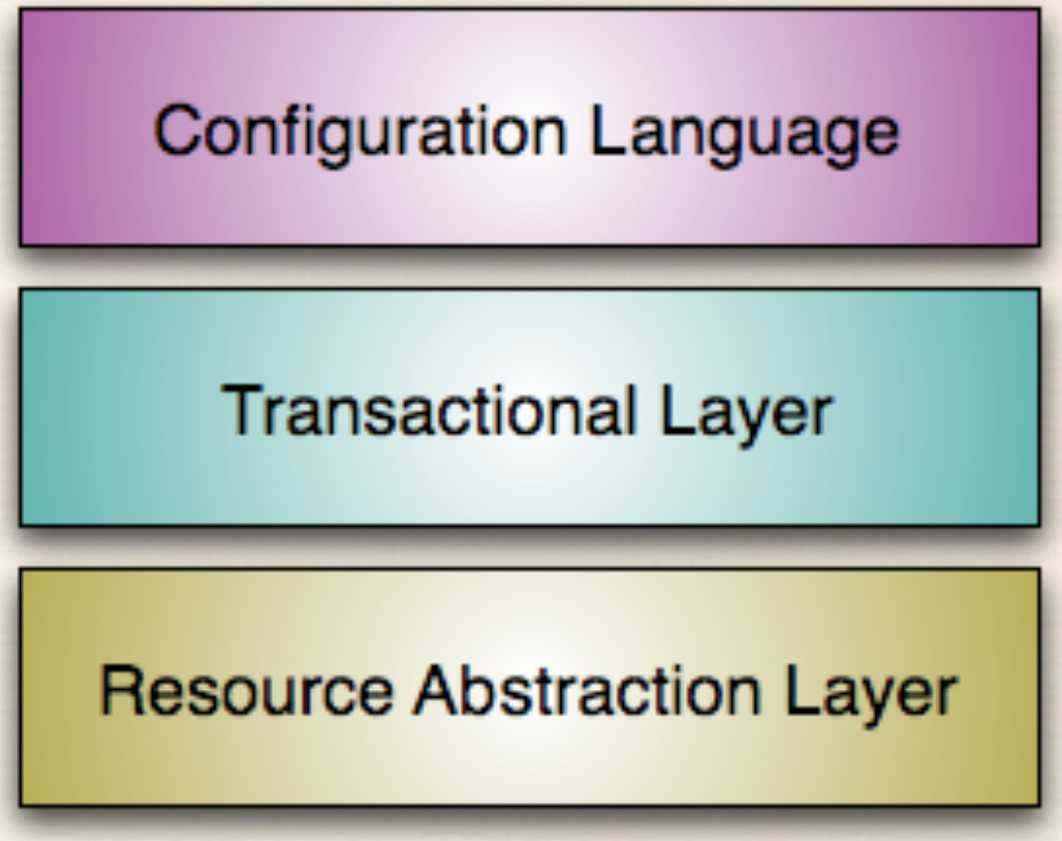
- 资源的抽象，一切皆资源
  - 资源是Puppet原语
- 明确资源间依赖关系
- 解释性配置语言

Features

- **Idempotency**
  - puppet can safely be run multiple times
- **Cross Platform**
  - RAL allows focus on system ignoring implementation details
- **Model & Graph Based**
  - resource modeled as type provider fulfilled the resource
  - graph based system modeling relationships between resources

Components

- Agent
- Facter
- ENC
- Transaction
- RAL
- Report



WorkShop

- 安装
- 认证
- 使用

生态圈

控制台 Dashboard | Foreman

配置管理PUPPET

命令编排MCollective

静态变量系统Facter

动态变量系统Hiera

数据服务 PUPPETDB