**Ansible中文权威指南**

**Ansible介绍:**

自动化运维就是将日常重复性的工作通过规则设定使其遵循预先既定规则，在指定的范围时间内自动化运行，但整个过程无需人工参与

Ansible其主要功能是帮助运维实现IT工作的自动化、降低人为操作失误、提高业务自动化率、提升运维工作效率，常用于软件部署自动化、配置自动化、管理自动化、系统化系统人为、持续集成、零宕机平滑升级等。它丰富的内置模块(acl、command、shell、cron、yum、copy、file、user等，多达569个)和开放的API接口

Ansible模块覆盖面的大致分类

**系统层：**支持的系统有Linux、Windows、AIX等。对应的模块又acl、cron、pip、easy\_install、yum、authorized\_key等大量的内置模块

**知名第三方平台支持：**支持的云平台有AWS、Azure、Cloudflare、Openstack、Google、Linode、Digital Ocean等，对应的模块有ec2、azure\_rm\_deployment、cloudflare\_dns、clc\_aa\_policy、glance\_image、gc\_storage、digital\_ocean等

**虚拟化：**VMware、Docker、Cloudstack、LXC、Openstack等，对应的模块又vmware\_vmkernel、docker、cs\_account、lxc\_container、glance\_image等

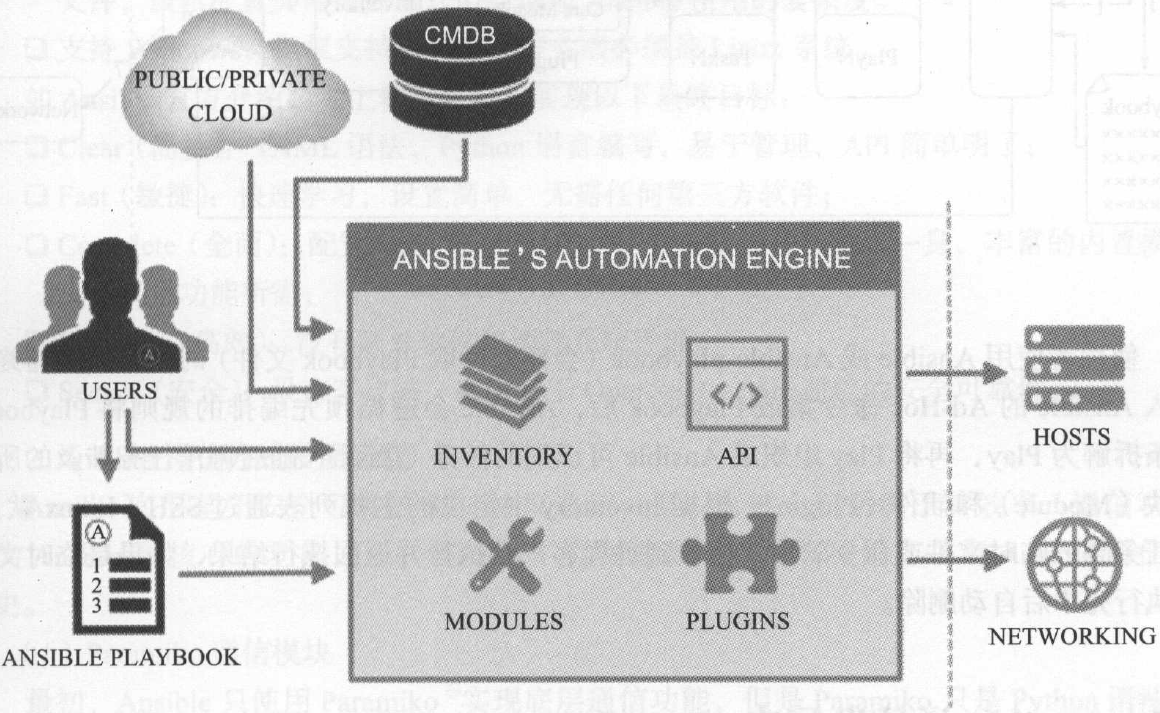
**商业化硬件：**F5、ASA、Citrix、Eos等，对应的模块又bigip\_facts、asa\_acl、netscaler、eos\_command等

**应用系统层：**Apache、Zabbix、Rabbitmq、SVN、GIT等，对应的模块又apache2\_module、zabbix\_group、rabbitmq\_binding、subversion、git等

**其他模块：**Cloud Modules(云主机模块)、Clustering Modules(集群模块)、Commands Modules(命令模块)、Database Modules(数据库模块)等

**Ansible工作模式**

Ansible没有客户端，因此底层通信依赖于系统软件，Linux系统下基于OpenSSH通信，Windows系统下基于PowerShell，管理端必须是Linux系统，使用者认证用过后在管理节点通过Ansible工具调用各应用模块将指令推送至被管理端执行，并在执行完毕后自动删除产生的临时文件。



**Ansible角色：**使用者、Ansible工具集、作用对象

**使用者：****Ansible使用者来源于多种维度**

**第一种方式：CMDB(配置管理数据库)**，CMDB存储和管理着企业IT架构中的各项配置信息。是构建IYIL项目的核心工具，运维人员可以组合CMDB和Ansible，通过CMDB直接下发指令调用Ansible工具集完成操作者所希望达成的目标

**第二种方式：PUBLIC/PRIVATE方式**，Ansible除了丰富的内置模块外，同时提供丰富的API语言接口。如PHP、Python、PERL等多种当下流行语言，基于PUBLIC(公有云)/PRIVATE(私有云)，Ansible以API调用的方式运行

**第三种方式：USERS直接使用Ad-Hoc临时命令集**调用Ansible工具集来完成任务执行

**第四种方式：USERS预先编写好的ANSIBLE PLAYBOOKS**，通过执行Playbooks中预先编排好的任务集按序完成任务执行

**Ansible工具集：Ansible命令是Ansible的核心工具**，Ansible命令并非自身完成所有的功能集，其只是Ansible执行任务的调用入口，大家可先理解为”总指挥”，所有命令的执行通过”调兵遣将”最终完成

**Ansible作用对象**

Ansible作用对象，不仅仅是Linux和非Linux操作系统的主机，同样也可以作用与各类公有云/私有云。商业和非商业设备的网络设备

**Ansible组成：**Ansible主要又6部分组成

**Ansible PLAYBOOKS：任务剧本(任务集)**，编排定义Ansible任务集的配置文件，由Ansible顺序依次执行，通常是JSON格式的YML文件

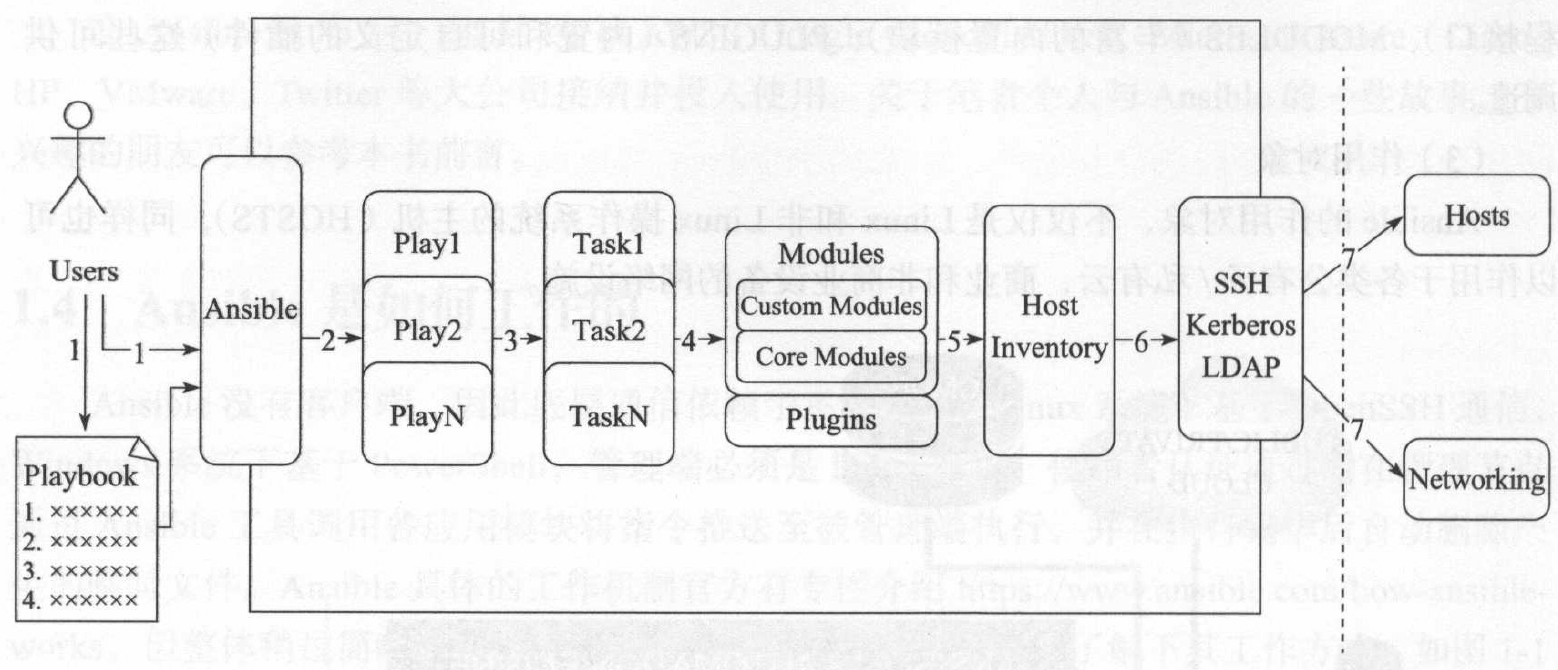
**INVENTORY：Ansible管理主机的清单**

**MODULES：Ansible执行命令的功能模块**，多数为内置的核心模块，也可自定义

**PLUGINS：模块功能的补充**，如连接类型插件、循环插件、变量插件、过滤插件等，该功能不常用

**API：供第三方程序调用的应用程序编程接口**

**Ansible：**该部分图中表示的不明显。组合INVENTORY、API、MODULES、PLUGINS的绿框大家可以理解为是Ansible命令工具，其为核心执行工具



使用者使用Ansible或Ansible-playbook(会额外读取Playbook文件)时，在服务器终端输入Ansible的Ad-Hoc命令集合Playbook后，Ansible会遵循预先编排的规则将Playbooks逐条拆解为Play，再将Play组织成Ansible可识别的任务(Task)，随后调用任务涉及的所有模块(Module)和插件(Pliugin)，根据Inventory中定义的主机了就通过SSH将任务集以临时文件或者命令的形式传输到远程客户端执行并返回执行结果，如果是临时文件则执行完毕后自动删除

**Ansible SHH工作机制**

Ansible执行命令时，通过其底层传输连接模块，将一个或数个文件，或者定义一个Play或Command命令传输到远程服务器/tmp目录的临时文件，并在远程执行这些Play/Comand命令，然后删除这些临时文件，同时回传整体命令执行结果。

**自动化工作类型归类：**

1. **应用部署：**现今的应用功能越来越强大，同步应用部署过程的依赖和规划也日趋复杂，但对应用运维的要求没有随之降低，有效快速正确平滑的应用部署需求日趋强烈。Ansible内置网络、应用、系统、第三方云平台扩展等完善的功能模块，协助运维快速完成应用的安装。卸载。升级、启动、配置等部署类工作，即使多跨平台或知名的商业硬件也同样支持
2. **配置管理：**配置管理是通过技术或行政手段对软件产品及其开发过程和生命周期进行控制。规范的一系列措施。配置管理的目标是记录软件产品的演化过程，确保软件开发者在软件生命周期中各个阶段都能得到精确的产品配置。在日益复杂的IT环境和用户需求下，Ansible内置File、Template，结合Jinja、Lineinfile等内置模块，同时无缝结合GitHub、GitLab、Git、SVN、Jenkins等主流版本控制和CI持续集成工具，助力配置管理自动化
3. **任务编排：**有效保障Tasks任务流安装既定规则和顺序完成时限制定的目标和计划，同时Roles编排方式又能在一定程度上从书写习惯和代码层编排上保证整体项目的可架构性和规范性，协助控制项目维护成本不致过高

如上场景适合网络管理员。系统运维、应用运维、桌面运维、DevOps、基础架构运维等多领域运维行业，以及无运维岗但服务规模又需有一定精力投入维护的小型公司，开发人员经过简单的理解即可初步上手。同样也适合于中大型公司，可以投入人力、精力、彩礼对Ansible进行二次开发，使其更加适应