

openstack™
CLOUD SOFTWARE

OpenStack从入门到精通实践 第十周

DATAGURU专业数据分析社区

【声明】 本视频和幻灯片为炼数成金网络课程的教学资料，所有资料只能在课程内使用，不得在课程以外范围散布，违者将可能被追究法律和经济责任。

课程详情访问炼数成金培训网站

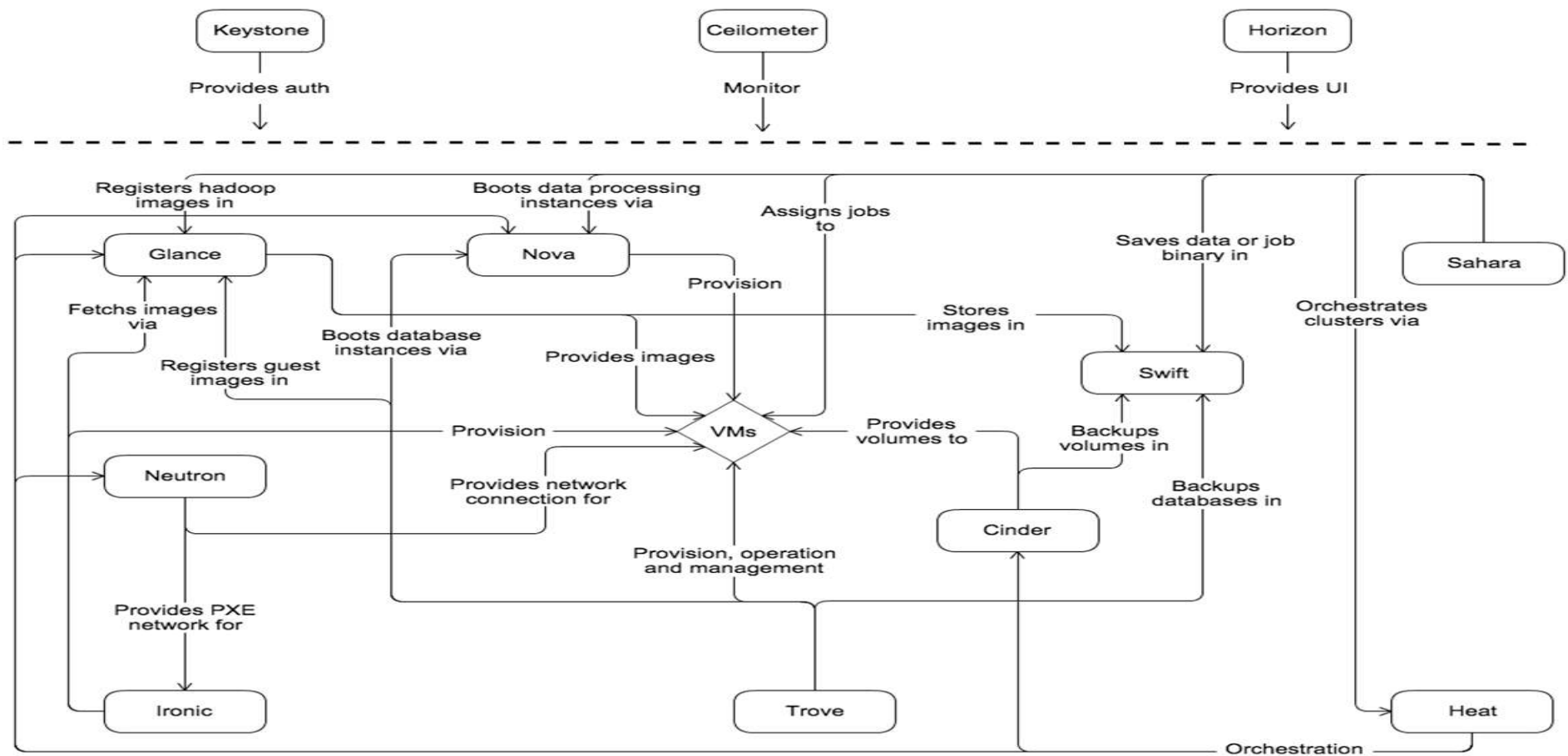
<http://edu.dataguru.cn>

炼数成金逆向收费式网络课程



- **Dataguru (炼数成金) 是专业数据分析网站，提供教育，媒体，内容，社区，出版，数据分析业务等服务。我们的课程采用新兴的互联网教育形式，独创地发展了逆向收费式网络培训课程模式。既继承传统教育重学习氛围，重竞争压力的特点，同时又发挥互联网的威力打破时空限制，把天南地北志同道合的朋友组织在一起交流学习，使到原先孤立的学习个体组合成有组织的探索力量。并且把原先动辄成千上万的学习成本，直线下降至百元范围，造福大众。我们的目标是：低成本传播高价值知识，构架中国第一的网上知识流转阵地。**
- **关于逆向收费式网络的详情，请看我们的培训网站 <http://edu.dataguru.cn>**

openstack总结回顾



Python语法（一）



- Python是一门动态语言
 - 不用编译,像脚本一样直接运行
 - 无需指定变量的类型,在运行时,根据它的内容来决定的类型
- Python以缩进来区分语句块
 - 同一层次的语句必须有相同的缩进
 - 缩进的方式包括tab和空格

```
49 def fail_gracefully(f):
50     """Logs exceptions and aborts."""
51     @functools.wraps(f)
52     def wrapper(*args, **kw):
53         try:
54             return f(*args, **kw)
55         except Exception as e:
56             LOG.debug(e, exc_info=True)
57
58             # exception message is printed to all logs
59             LOG.critical(e)
60             sys.exit(1)
61
62     return wrapper
```

■ 函数

- 使用def关键字定义函数，函数包括函数名、括号、括号中的参数和冒号，不需要定义返回类型
- 函数参数除了正常定义的必选参数外，还可以使用默认参数、可变参数和关键字参数
 - 函数定义：`def func(a, b, c=0, *args, **kw):`
 - 函数调用：`func(1, 2, 3, 'a', 'b', x=99)`
 - 函数输出：`a = 1 b = 2 c = 3 args = ('a', 'b') kw = {'x': 99}`
- 返回值可以返回0个或1个或多个，类型可以是各种类型
- 函数名也是变量,一个函数就可以接收另一个函数作为参数
- Python使用lambda关键字定义匿名函数。即，函数没有具体的名称,而用def创建的方法是有名称的
 - `def add(x, y):`
`return x + y`
 - `lambda x, y: x + y`

Python语法（三）



■ 模块

- 一个文件后缀名为.py的文件就可以称作一个模块
- 通过包来组织模块，避免冲突。每一个包目录下面都会有一个__init__.py的文件
- 通过import或者from...import来导入相应的模块
 - import sys
 - from sys import argv,path
- 第三方的模块可以通过pip来安装
- 模块的搜索路径存放在sys模块的path变量中，可以通过sys.path.append或PYTHONPATH来增加

```
>>> import sys
>>> sys.path
['', '/opt/stack/keystone', '/opt/stack/glance', '/opt/stack/cinder', '/opt/stack/neutron', '/opt/stack/nova', '/opt/stack/horizon', '/usr/lib/python2.7', '/usr/lib/python2.7/plat-x86_64-linux-gnu', '/usr/lib/python2.7/lib-tk', '/usr/lib/python2.7/lib-old', '/usr/lib/python2.7/lib-dynload', '/usr/local/lib/python2.7/dist-packages', '/usr/lib/python2.7/dist-packages']
```

virtualenv和tox



■ virtualenv

- 用于创建独立的Python环境，多个Python相互独立，互不影响
- 常见命令
 - `virtualenv venv` #创建虚拟环境
 - `source ./bin/activate` #启动虚拟环境
 - `deactivate` #退出虚拟环境

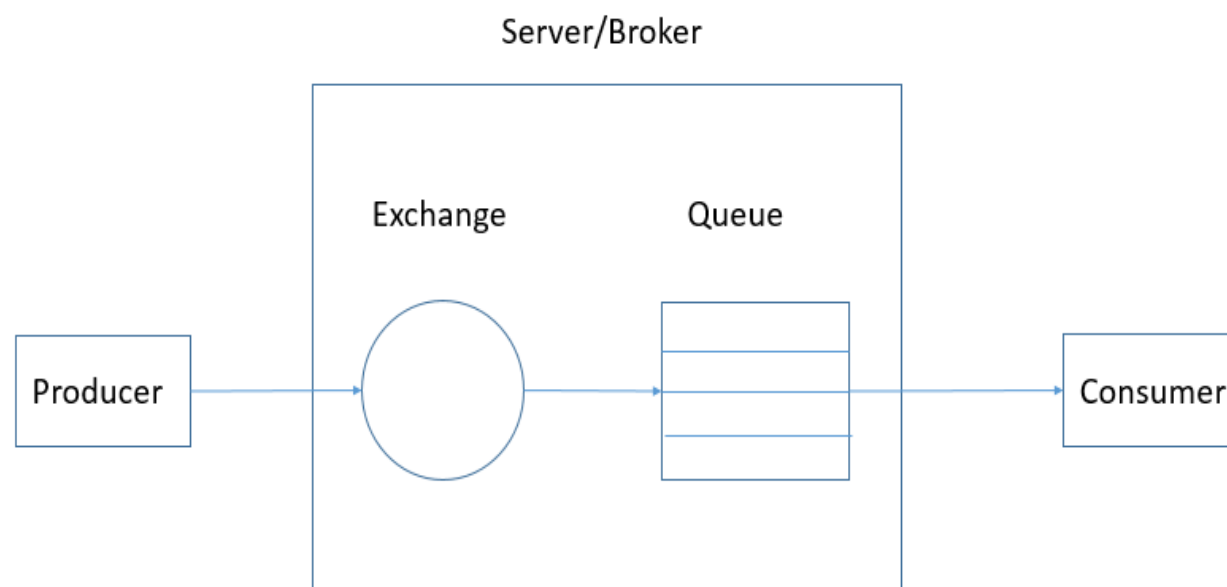
■ tox

- 虚拟环境管理和测试命令行工具
- 配置文件tox.ini

openstack消息总线（一）



- AMQP:是一个异步消息传递所使用的开放的应用层协议规范
- AMQP的主要参与者
 - Producer：消息的产生者
 - Consumer：消息的接收者
 - Exchange：交换部件，根据消息的条件选择不同的消息接收者。
 - Queue：消息队列，暂时缓存到达消费者的消息
 - Server/Broker：实现了AMQP的中间件服务

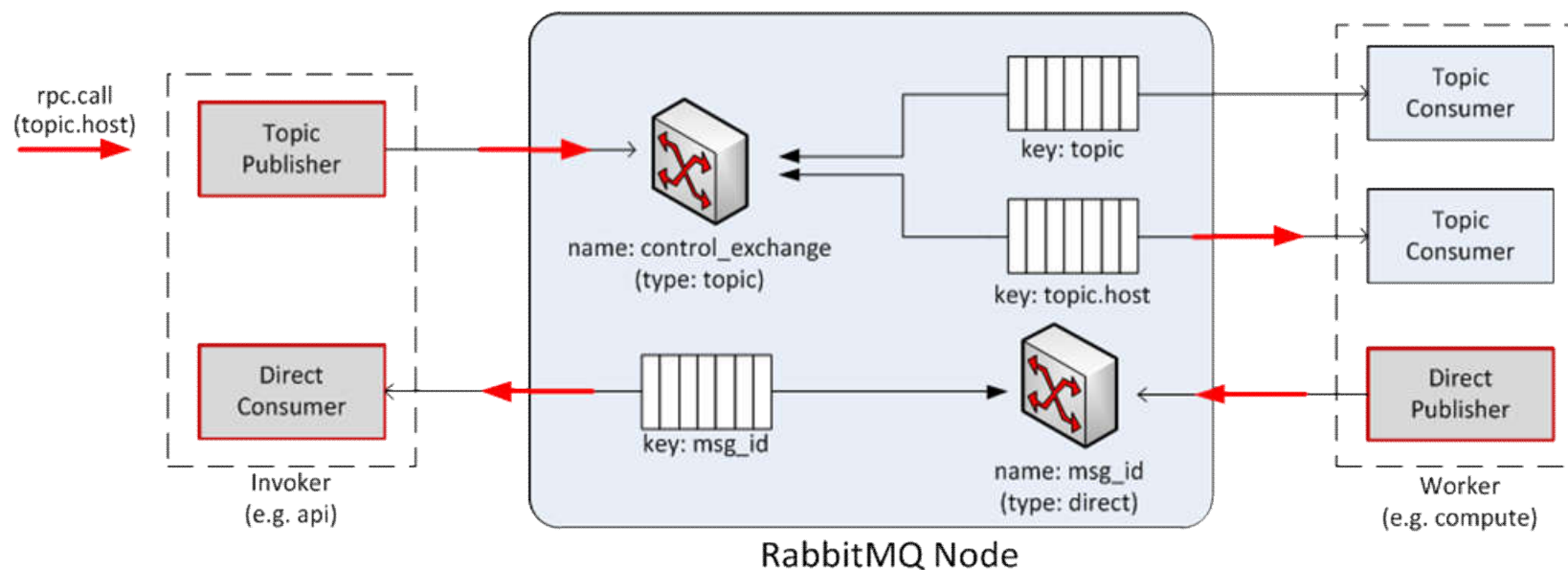


- AMQP 定义了三种类型的 Exchange，不同类型 Exchange 实现不同的 routing 算法
 - Direct Exchange：Point-to-Point 消息模式，消息点对点的通信模式，Direct Exchange 根据 Routing Key 进行精确匹配，只有对应的 Message Queue 会接受到消息
 - Topic Exchange：Publish-Subscribe(Pub-sub)消息模式，Topic Exchange 根据 Routing Key 进行模式匹配，只要符合模式匹配的 Message Queue 都会收到消息
 - Fanout Exchange：广播消息模式，Fanout Exchange 将消息转发到所有绑定的 Message Queue

openstack消息总线（三）

■ RPC.call

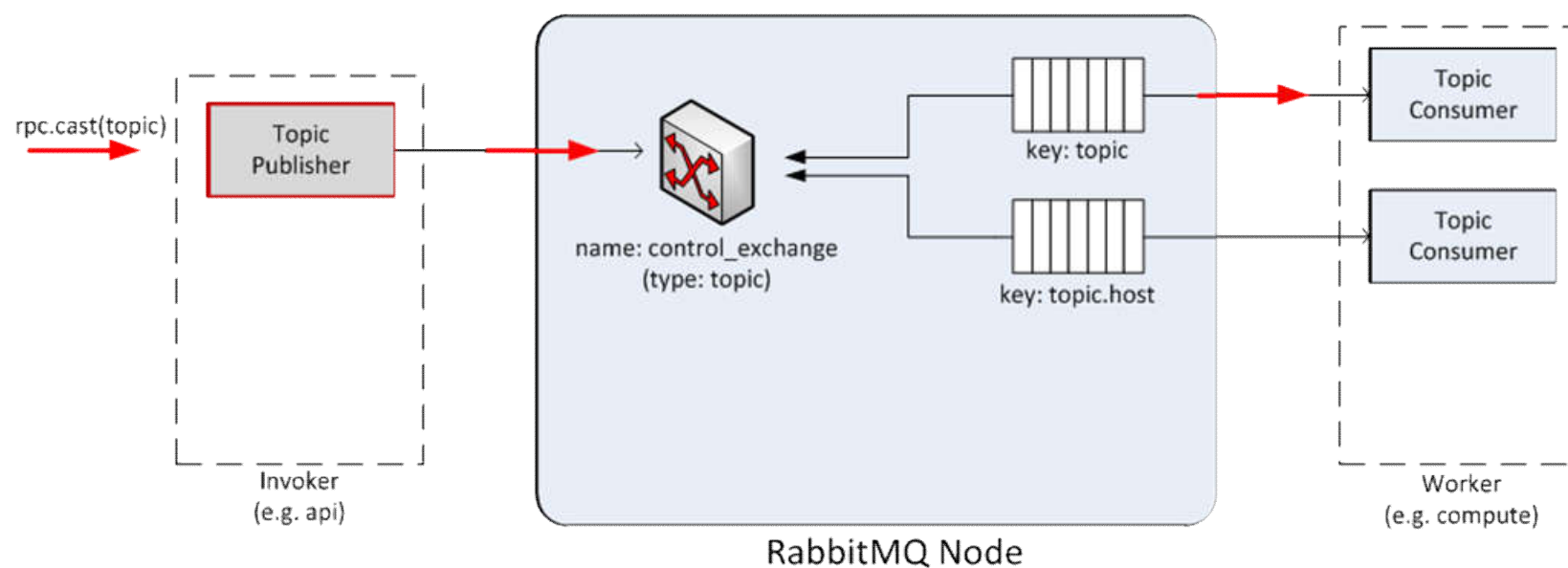
- 发送 RPC 请求，并返回请求处理结果



openstack消息总线（四）

■ RPC.cast

- 发送rpc请求，无返回



■ 目的

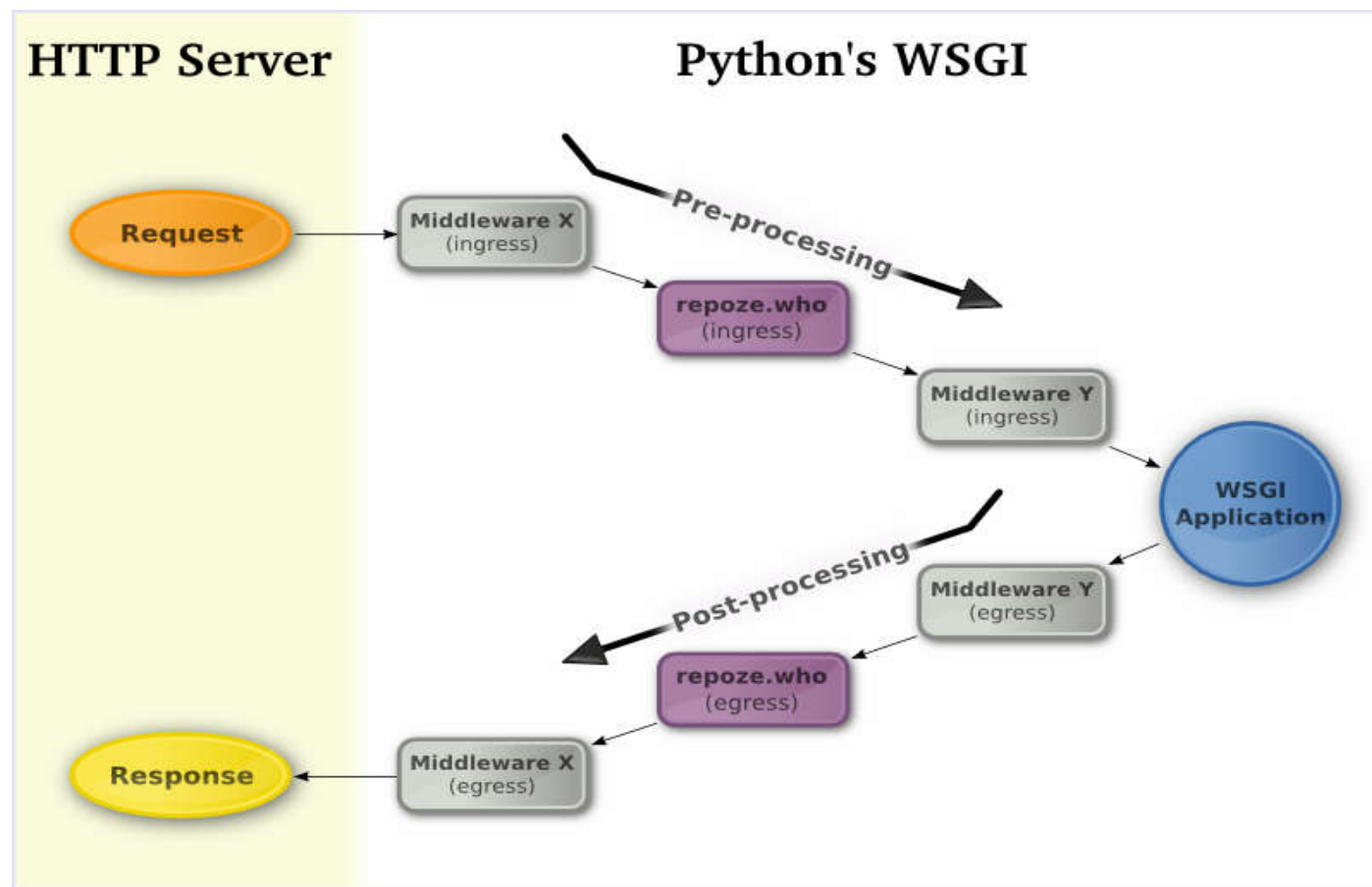
- 将项目代码打包成源码包或者二进制包进行分发

■ 打包工具

- setuptools和pbr
 - yum install python-setuptools
- 制作分发包
 - python setup.py sdist #制作源代码分发包
 - python setup.py bdist #制作编译文件分发包，支持tar.gz,rpm,exe,egg等格式
- 安装包
 - python setup.py install
 - easy_install xxxx.egg

WSGI

- WSGI 是 Python 应用程序或框架与 Web 服务器之间的一种接口
 - web 服务器
 - web 应用程序
 - web 中间件



- GET (SELECT) : 从服务器取出资源 (一项或多项) 。
- POST (CREATE) : 在服务器新建一个资源。
- PUT (UPDATE) : 在服务器更新资源 (客户端提供改变后的完整资源) 。
- PATCH (UPDATE) : 在服务器更新资源 (客户端提供改变的属性) 。
- DELETE (DELETE) : 从服务器删除资源。
- HEAD : 获取资源的元数据。
- OPTIONS : 获取信息 , 关于资源的哪些属性是客户端可以改变的。

状态码（一）



- 200 OK - [GET]：服务器成功返回用户请求的数据，该操作是幂等的（Idempotent）。
- 201 CREATED - [POST/PUT/PATCH]：用户新建或修改数据成功。
- 202 Accepted - [*]：表示一个请求已经进入后台排队（异步任务）
- 204 NO CONTENT - [DELETE]：用户删除数据成功。
- 400 INVALID REQUEST - [POST/PUT/PATCH]：用户发出的请求有错误，服务器没有进行新建或修改数据的操作，该操作是幂等的。
- 401 Unauthorized - [*]：表示用户没有权限（令牌、用户名、密码错误）。
- 403 Forbidden - [*] 表示用户得到授权（与401错误相对），但是访问是被禁止的。
- 404 NOT FOUND - [*]：用户发出的请求针对的是不存在的记录，服务器没有进行操作，该操作是幂等的。

状态码（二）



- 406 Not Acceptable - [GET]：用户请求的格式不可得（比如用户请求JSON格式，但是只有XML格式）。
- 410 Gone -[GET]：用户请求的资源被永久删除，且不会再得到的。
- 422 Unprocesable entity - [POST/PUT/PATCH] 当创建一个对象时，发生一个验证错误。
- 500 INTERNAL SERVER ERROR - [*]：服务器发生错误，用户将无法判断发出的请求是否成功。

openstack SDK



- openstack软件开发包
 - <https://wiki.openstack.org/wiki/SDKs>

C++

Java

.Net

Python

PHP

Ruby

- 用途
 - 操作openstack服务的api接口
- API清单
 - python-keystoneclient
 - python-glanceclient
 - python-novaclient
 - python_neutronclient
 - python-cinderclient
 - python-swiftclient

Thanks

FAQ时间