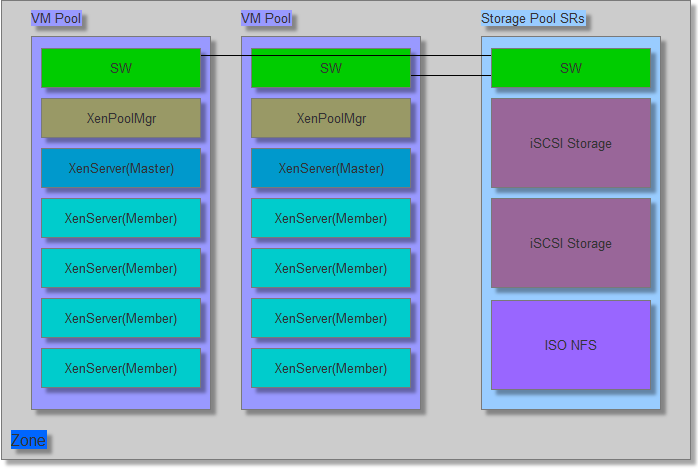
Xen云平台设计文档

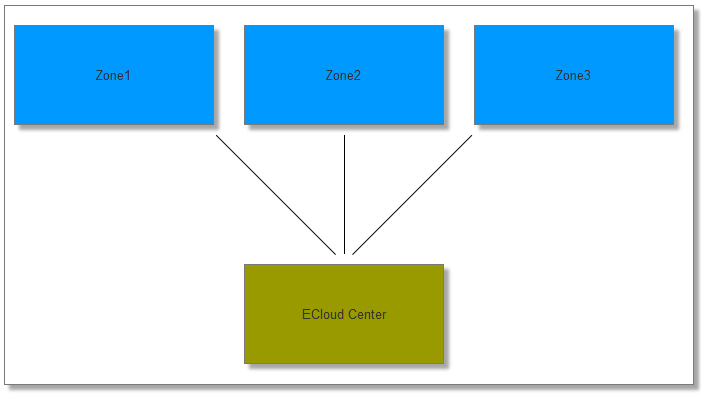
# 逻辑架构图

整个云平台以机房，资源池（虚拟池，存储池）为组成单元，一个机房由多个资源池构成，其中资源池包含虚拟池和存储池，虚拟池主要运行虚拟机实例，存储池主要是为虚拟机实例提供虚拟磁盘，ISO镜像等存储服务，整个云平台以多个机房组成，具体逻辑关系可以参考下面的图。

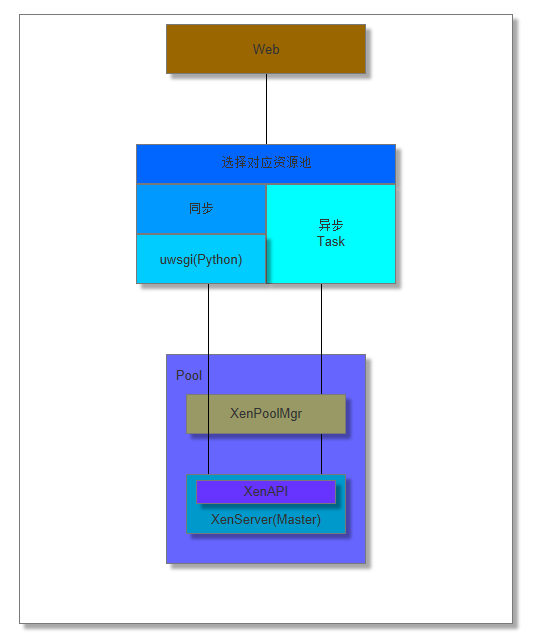
## 机房逻辑单元图



## 云平台概要图



## API逻辑架构



整个API接口使用HTTP机制，传送数据使用json编码，具体执行接口根据实际情况分同步和异步两种。同步通常是耗时比较低的，比如获取主机，虚拟机列表等，同步方式是通过uswgi使用python直接调用对应资源池的XenServer Master主机上面的XenAPI；异步主要是一些比较耗时的，比如创建虚拟机，创建磁盘，开机关机等，具体以Task任务的方式提交，然后由指定的XenPoolMgr获取任务，然后在调用所在资源池的Master主机XenAPI，最后把调用执行结果提交上Task，Web端再通过轮训的方式获取Task执行结果。

# 基础数据结构

## 资源池数据结构

|  |  |  |
| --- | --- | --- |
| 字段 | 类型 | 描述 |
| id | int | 资源id |
| name | string | 资源池名称 |
| type | string | 资源池类型  vm=虚拟机池  storage=存储池 |
| sub\_type | string | 子类型 |
| status | string | 资源状态  enable=可用  disable=不可用 |
| zone | string | 资源池所在机房 |
| rack | string | 资源池所在机柜 |
| data | string | 资源池具体扩展信息json结构  主要是池的主机地址，账号，接口等 |
| Desc | string | 资源池描述 |

# 基础API

## 资源池API

|  |  |
| --- | --- |
| URL | 说明 |
| 获取当前资源池列表  <http://61.142.208.98/ecloud_vm_api/webserver/pool.php> | POST数据  **data={"opt":"get\_pool\_list","type":"xxx"}**  type=all 获取所有资源池列表  type=vm 获取虚拟资源池类型列表  type=storage 获取存储资源池类型列表 |
| 返回json数据 |
| 获取指定资源池主机列表  <http://61.142.208.98/ecloud_vm_api/webserver/pool.php> | POST数据  **data={"opt":"get\_pool\_host\_list","pool\_name":"资源池名称"}** |
| 返回json数据 |
|  |  |
|  |  |
|  |  |

## 主机API

## 虚拟机API

# 任务API

## 任务数据结构

|  |  |  |
| --- | --- | --- |
| 字段 | 类型 | 描述 |
| id | int | 自增任务id |
| totype | string | 指定执行该任务所属类型  pool = 资源池 |
| todo | string | 指定执行该任务的具体单元  比如totype = pool，则todo=资源池名 |
| user | string | 提交任务的用户名 |
| type | string | 任务类型 |
| sub\_type | string | 子类型 |
| submit\_time | timestamp | 提交任务时间 |
| data | String | 任务具体描述json格式 |
| status | string | 任务状态  init=初始化就绪  dong=正在执行  finish=执行完毕 |
| start\_time | timestamp | 任务开始执行时间 |
| finish\_time | timestamp | 任务执行完毕时间 |
| ret | int | 执行返回值 0成功 非0识别 |
| result | string | 任务执行结果json格式 |
| error | string | 任务执行错误信息  (如果ret!=0) |
|  |  |  |

都是POST JSON格式的请求数据，JSON数据作为BODY POST到指定接口，BODY使用”data”字段作为提交。比如 data=JSON

## 提交任务

php代码

|  |
| --- |
| $task\_data = array  (  [arg1] => 1  [arg2] => 2  );  $task = array(  'opt' => 'submit',//操作类型  'totype' => 'totype',//任务类型  'todo' => 'todo',//任务类型  'type' => 'test\_type',//任务类型  'sub\_type' => 'test\_sub\_type',//任务子类型  'user' => 'test',//提交任务的用户名  'data' => json\_encode($task\_data)//任务具体描述信息json编码  );  json\_encode($task); |

转换成要提交的json数据：

|  |
| --- |
| {"opt":"submit","type":"test\_type","sub\_type":"test\_sub\_type","user":"test","data":"{\"arg1\":1,\"arg2\":2}"} |

提交任务返回结果php代码：

|  |
| --- |
| $result = array(  'ret' => 0,//返回码，0成功 非0失败  'result' => taskid,//如果提交成功，则这里返回任务id  'error' => ‘’//如果提交失败，则这里返回具体错误信息  );  json\_encode($result); |

转换成要返回的json数据：

|  |
| --- |
| {"ret":0,"taskid":taskid} |

## 删除任务

php代码

|  |
| --- |
| $task = array(  'opt' => 'delete',//操作类型  'taskid' => taskid,//任务id  );  $json = json\_encode($task); |

转换成要提交的json数据：

|  |
| --- |
| {"opt":"delete","taskid":taskid} |

删除任务返回结果php代码：

|  |
| --- |
| $result = array(  'ret' => 0,//返回码，0成功 非0失败  'result' => taskid,//如果删除成功，则这里返回任务id  'error' => ‘’//如果删除失败，则这里返回具体错误信息  );  json\_encode($result); |

转换成要返回的json数据：

|  |
| --- |
| {"ret":0,"taskid":taskid} |

## 设置任务

php代码

|  |
| --- |
| //把想要设置的字段与对应的值都写紧’set’字段里面  //比如要把taskid=xxx的任务status字段设置为finish，ret字段为0  $task\_set = array('status' => 'finish', 'ret' => 0);  $task = array(  'opt' => 'set',  'taskid' => taskid,  'set' => json\_encode($task\_set)  );  json\_encode($task); |

转换成要提交的json数据：

|  |
| --- |
| {"opt":"set","taskid":taskid,"set":"{\"status\":\"finish\",\"ret\":0}"} |

设置任务返回结果php代码：

|  |
| --- |
| $result = array(  'ret' => 0,//返回码，0成功 非0失败  'result' =>taskid,//如果设置成功，则这里返回任务id  'error' => ‘’//如果设置失败，则这里返回具体错误信息  );  json\_encode($result); |

转换成要返回的json数据：

|  |
| --- |
| {"ret":0,"taskid":taskid} |

## 查询任务

php代码

|  |
| --- |
| $task = array(  'opt' => get,  'taskid' => 123,//要查询指定taskid的，如果taskid为空，则查询所有任务  //这里可以再增加其他查询条件，比如totype, todo之类的，  //  );  print(json\_encode($task)); |

转换成要提交的json数据：

|  |
| --- |
| {"opt":"get","taskid":taskid} |

设置任务返回结果php代码：

|  |
| --- |
| $result = array(  'ret' => 0,//返回码，0成功 非0失败  'result' => array(),//如果查询成功，则这里返回符合条件的任务列表  'error' => ‘’//如果查询失败，则这里返回具体错误信息  );  json\_encode($result); |

转换成要返回的json数据：

|  |
| --- |
| {"ret":0,"tasks":[],"error":""} |