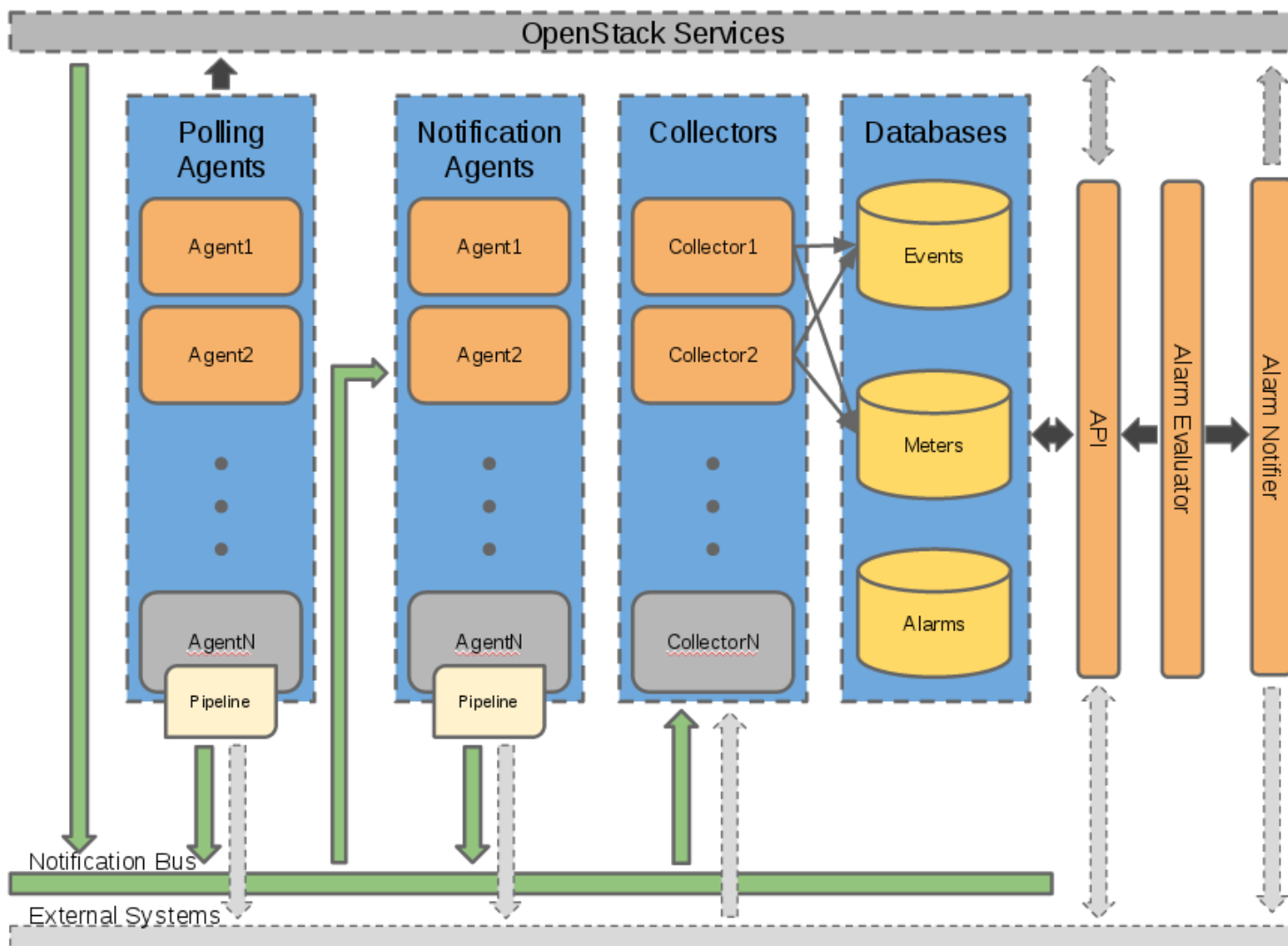


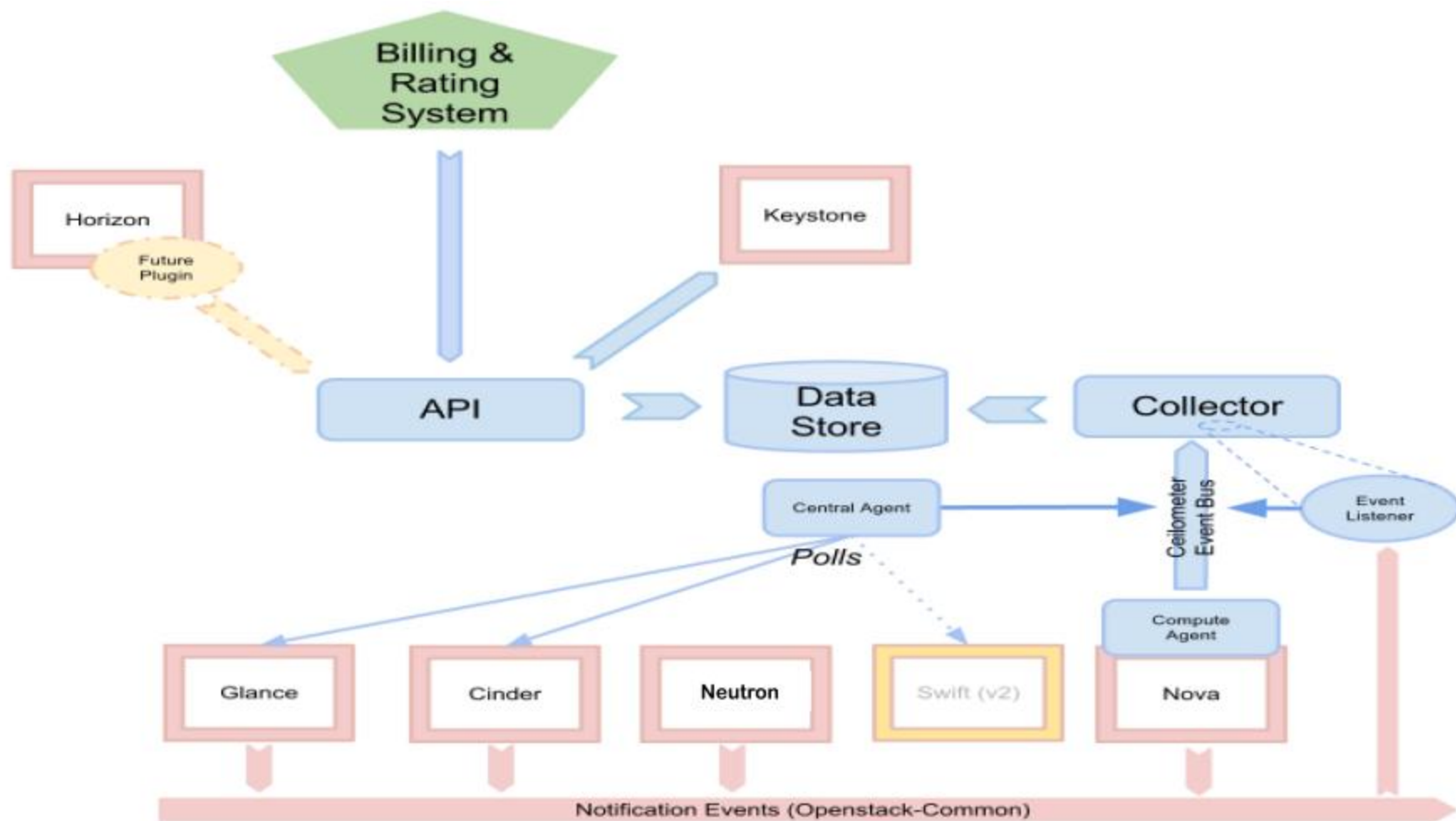
# Ceilometer

- Ceilometer项目开始于2012年，最初用于一个简单目标：收集OpenStack项目的信息，计费系统使用该数据源生成费用账单，称之为“计量”。
- 随后，Ceilometer收集的指标越来越多，社区开始给Ceilometer增加第二个目标：成为一个标准的采集指标机制，而不管指标的用途。
- 最近，随着Heat项目的诞生，OpenStack项目需要一个工具来观察关键变量，并触发不同的响应。因为Ceilometer已经收集了大量的指标，该工作顺理成章的成为了Ceilometer的扩展，称之为“告警”。
  - ✓ 注：从L版本开始，Ceilometer项目专注于指标采集。告警从Ceilometer独立出来，创建了一个新的项目Aodh。

# Ceilometer架构



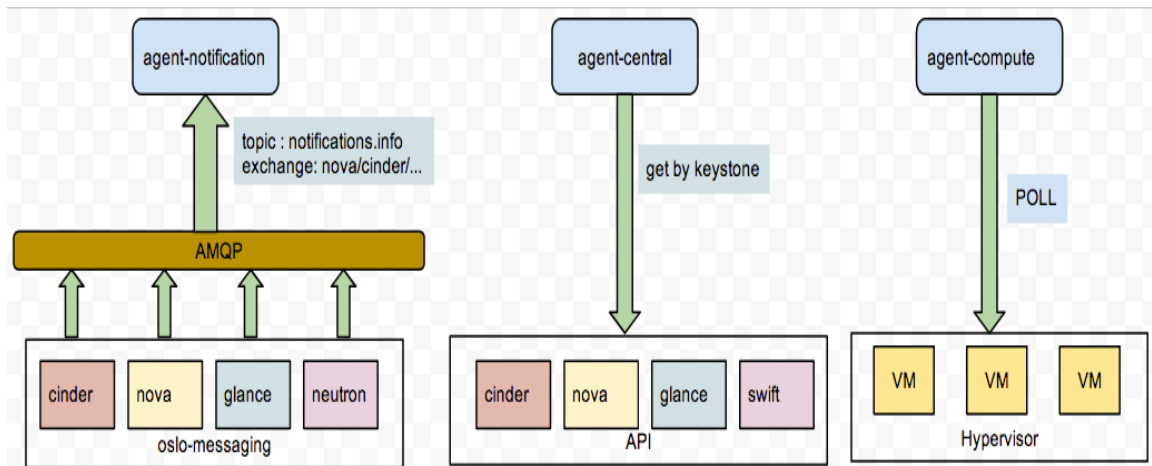
# Ceilometer架构



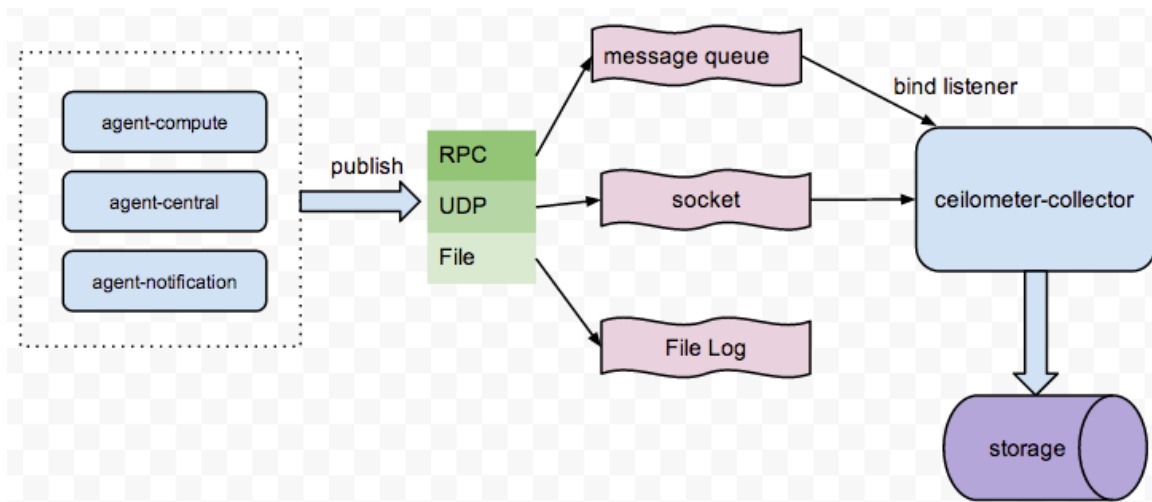
# Ceilometer基础组件

- Compute Agent
  - 运行在每个计算节点，采集性能指标
- Central Agent
  - 运行在管理节点，调用OpenStack其它组件api采集指标
- Notification Agent
  - 运行在管理节点，接收其它组件主动上报的通知消息
- Collector
  - 运行在管理节点，基于AMQP接收消息，并记录到Data Store
- Data Store
  - 持久化数据存储 ( MongoDB/MySQL )
- API
  - 运行在管理节点，提供接口访问Data Store
  - 设置告警条件和动作
- Alarm Evaluator
  - 运行在管理节点，根据用户设置的告警条件进行告警评估
- Alarm Notifier
  - 运行在管理节点，告警通知 ( Http callback/ LOG )

# Ceilometer数据采集机制



- agent-notification负责收集各个组件推送的消息
- agent-central是通过各个组件API方式收集有用的信息
- agent-compute只负责收集本计算节点信息



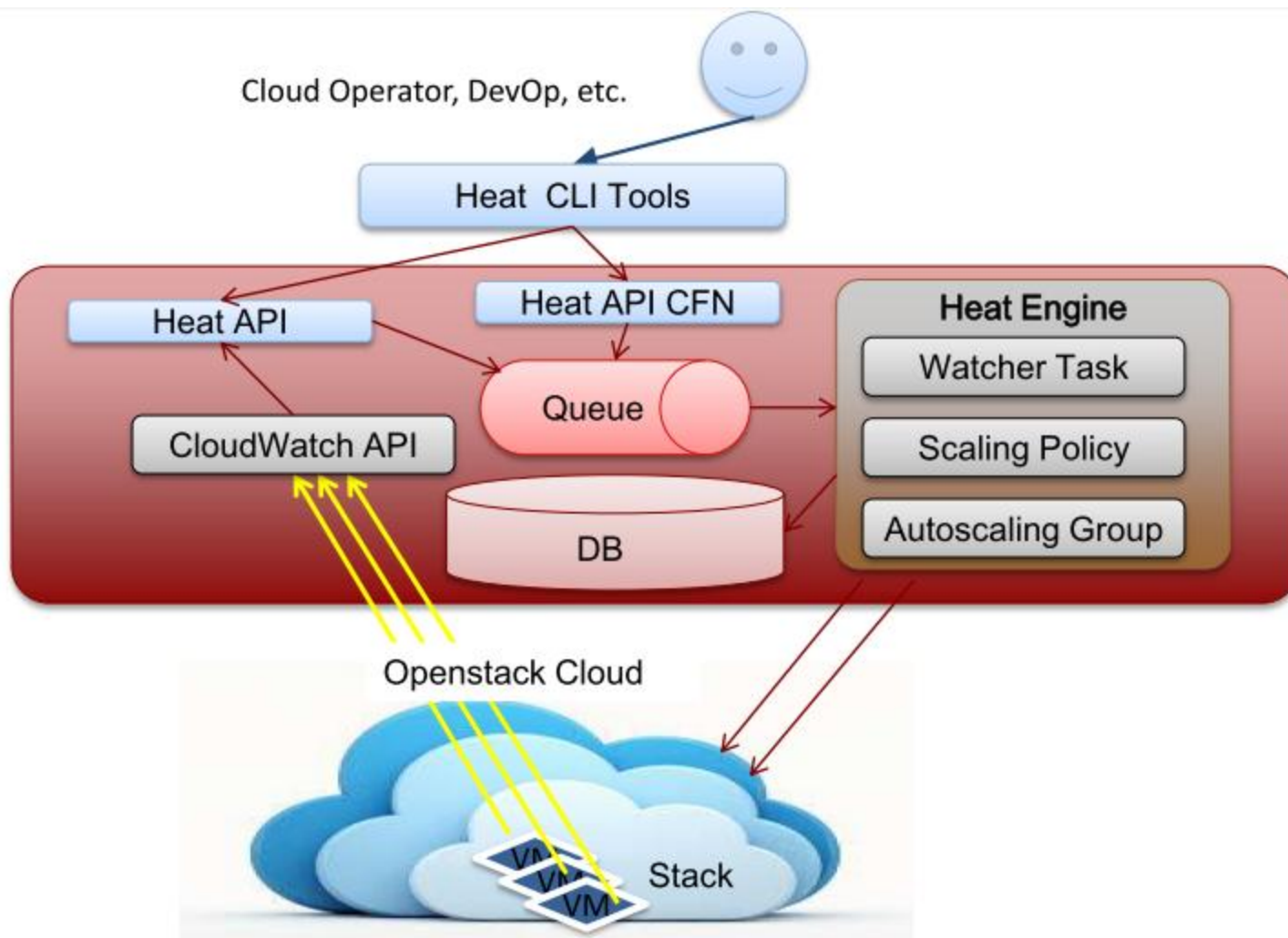
agent-\*服务负责采集信息，采集的信息可以通过三种方式发布：

- RPC
- UDP
- File

# Heat

- Heat提供了一个模板来描述云应用，通过调用相应的OpenStack API接口生成运行的云应用程序。
- Heat将其它OpenStack的核心组件集成到了一个文件模板系统，该模板允许创建大部分OpenStack资源类型（例如实例、浮动IP、卷、安全组、用户等），以及一些更高级的功能例如高可用性、自动缩放和嵌套栈。通过与其它OpenStack核心项目的紧密集成，所有OpenStack核心项目能够获得更大的用户群。
- 允许部署人员直接通过Heat集成，或者增加自定义插件。

# Heat架构



# Heat基础组件

- heat
  - heat工具是一个CLI，与heat-api通信来执行AWS CloudFormation API。终端开发者也可以直接使用heat REST API。
- heat-api
  - 提供OpenStack原生REST API，将API请求通过RPC发送给heat-engine。
- heat-api-cfn
  - 提供AWS查询API，与AWS CloudFormation兼容，将API请求通过RPC发送给heat-engine。
- heat-engine
  - 主要职责是协调模板的启动，并提供事件返回给API用户。
- heat-cfntools
  - 包括了一些和Heat配合使用的小工具，它们运行在虚拟机内部。



# Heat实现的功能

- 云基础设施搭建
  - 描述云基础设施的资源，例如服务器、网络、存储等，以及它们之间的关联
  - 修改云平台部署并实施
- 虚拟机内软件部署
  - 与Puppet和Chef集成
  - 通过cloud-init传递参数
- 弹性伸缩
  - 基于Ceilometer告警机制实现资源弹性伸缩
- 虚拟机HA
  - 监控虚拟机内部软件状态
  - 定义软件异常处理策略，重启服务/进程或虚拟机