A bill for cloud services is created by applying a rate plan to the metered data consumed by the customer.

This document provides an overview of the metered data and rate plan options for billing and showback of cloud services in OpenStack using OpenBook from Talligent.
Table of Contents

ELEMENTS OF AN INVOICE ........................................................................................................ 3
STEP 1 – METERING .................................................................................................................. 4
STEP 2 – RATING ...................................................................................................................... 4
STEP 3 – BILLING ..................................................................................................................... 5
DATA FOR BILLING .................................................................................................................. 5
SUMMARY .................................................................................................................................. 7
Billing for OpenStack Cloud Services

IT service delivery is changing rapidly, as are the needs and expectation of customers and providers alike. Cloud services range from finely metered a la carte offerings to complex packaged solutions, and will no doubt continue to evolve considerably over the next 12 to 24 months. On-demand, self-service delivery of cloud services requires controls to ensure the efficient distribution of scarce resources. These controls take the form of resource quotas, management reports (including showback), and bills for service. This document provides an overview of cloud billing and showback options in OpenBook.

Many public cloud providers start by providing services that are similar to AWS, the cloud model that most people are familiar with. To drive customer adoption and create higher value, public cloud providers differentiate by adding managed services, prepackaged applications, or increased security. It is important to be able to track and provide a rate for these differentiated attributes.

Private cloud administrators, on the other hand, generally have 2 main goals: 1) demonstrate that they can provide equivalent private cloud services at a similar or cheaper price than AWS, mitigating the effect of Shadow IT; and 2) encourage good behavior (and high utilization) by cloud tenants. Private cloud administrators generally don’t try to recoup all costs associated with their cloud, so payment processing is not needed, but integration with a cost accounting solution can be useful for creating budgets and collecting soft dollars.

Whether you are deploying a private or public cloud, OpenBook from Talligent has the flexibility to define cloud billing models that range from simple flat-rate billing to highly sophisticated packaging and pricing models.

Elements of an Invoice

If we look at the process model used by the Ceilometer team, there are three steps to constructing a bill for cloud services:

1. **METERING** – This step includes identifying billable features such as resource elements (CPU, memory, storage, bandwidth), applications, services, and other attributes; and collecting information about how much and often that feature was invoked.

2. **RATING** – Creating a monetary rate measured per event or time interval such as hour, day, or month and applying it to the metered data.

3. **BILLING** – The final bill is created when all price promotions, discounts, credits, taxes and currency conversions are applied.
**Step 1 – Metering**

The Ceilometer project within OpenStack is tasked with creating an infrastructure to collect all pertinent information about OpenStack projects and provide that information through a single source. Targeted uses for this information include billing, reporting, and monitoring. To quote the Ceilometer documentation, “Ceilometer's initial goal was, and still is, strictly limited to step one.” At this point, there is no movement for Ceilometer to take on the additional roles of rating or billing for the cloud.

OpenBook captures a full set of configuration details and usage metrics from OpenStack via the Ceilometer module. This includes relevant meters associated with Nova (Compute), Neutron (Network), Glance (Image), Cinder (Volume), and Swift (Object Storage). As new metrics are added to Ceilometer, those metrics are picked up by the resource manager in OpenBook and made available as billable elements included in rate plans. For example, the Juno release of OpenStack will add key SDN metrics such as Load Balancer as a Service, Firewall as a Service, and VPN as a Service. Those SDN metrics will then be available in OpenBook within a few weeks of the GA release of the Juno version of OpenStack.

This information is stored independently from Ceilometer to create a detailed historical record of your OpenStack cloud. Many metrics are not relevant to billing, but are important for other OpenBook features such as enterprise reporting and capacity planning. Since OpenBook uses the standard Ceilometer APIs, there is no vendor lock-in to a particular Openstack distribution or release.

From a billing and chargeback standpoint, OpenBook lets you apply rates on metered data such as:

- Virtual machines and their specific components such as vCPUs and memory
- Infrastructure elements such as networks, storage, and their configurations
- Cloud applications and middleware, including virtual desktop infrastructure
- Operating system – upcharge for Windows or enterprise Linux
- User defined features that can represent attributes or services not captured in OpenStack such as backup/recovery, SLA, bandwidth guarantees, etc.

User defined features are created and stored within OpenBook.

**Step 2 – Rating**

Rating is the process of applying a rate plan to the metered data. Administrators may want to launch their cloud with a relatively simple and straightforward rate plan so that customers understand what they are paying for and how it is being metered. In chargeback and showback situations, it may be enough to bill on the simple existence of a feature in a month. For on-demand public clouds, the rate plan will generally be much more granular, taking into account hourly or partial hour charges, although reserved instances allow users to prepay for servers on an annual basis (or longer).

Here are a few examples of rate plans in OpenBook:

- Simple existence of an element and its attributes
To create a final bill, whether for invoicing, chargeback, or showback reporting, all discounts, credits, taxes, and other adjustments are applied. Usage-based discounts are typically deployed via rate plan tiers in the previous step, but in order to support field sales flexibility, additional discounts can be offered at the customer or reseller level and tracked in OpenBook.

Prepaid credits lower the account receivable risk to the cloud provider and lower the risk of cost overruns to the cloud tenant. Credits are applied and drawn down by event- and usage-based costs. Tenants are notified when certain thresholds are reached, and can track balances and replenish credits through OpenBook. OpenBook can even be configured to revoke access and delete instances automatically when balances fall behind.

Final billing information can be passed via OpenBook’s Rest APIs into third party invoicing solutions so that a consolidated bill can be presented to the customer. Alternatively, this billing information can be passed to third party accounting solutions in the case of chargeback and internal budget tracking.

The list of pricing options includes:

- Discounts
- Promotions
- Pro-rated charges
- Localization, including currency and local taxation

When it comes to marketing your cloud offering, OpenBook provides all of the typical pricing and sales promotion mechanisms required to attract new customers, reward existing customers, and support a diverse customer base.

### Data for Billing

This section examines the various data elements and how they can be assembled into a meaningful bill for OpenStack services. As mentioned above, OpenBook captures a full set of configuration details and usage metrics from OpenStack via the Ceilometer module. Many details are useful for reporting and analysis of tenant workloads but are not relevant for billing. Billable elements generally fall into 4 categories:

1. Provisioned features such as entity types, states, and attributes available from Ceilometer and other OpenStack components;
2. Usage based features such as bandwidth available from Ceilometer;

3. Derived metrics that have to be calculated by combining usage-based metrics and configuration details

4. User defined attributes not available in Ceilometer.

Features are either pre-configured or user-defined. Pre-configured features are defined by OpenStack and are automatically populated into the OpenBook environment upon initial configuration. Each feature includes a vendor, feature type, and available subtypes. Subtypes are different varieties of a specific feature and will pre-populate the rate plan wizard in OpenBook. Multiple subtypes can be created for any feature. Features are billed based on state and not based on usage volume.

The following diagram shows the three states available for billing – In Existence, Powered On, In Use – as well as the OpenStack actions that control the state of an instance. As a point of reference, large public cloud vendors start charging for an instance once it has been provisioned for a user, regardless of whether the user is logged in or not. An instance can be stopped so that changes can be made. Instance charges do not accrue in the stopped state until the instance is started (a process to boot the instance). Storage charges may still apply in the stopped state. The equivalent in OpenStack would be the Powered On state, rather than the In Use state.

Usage based metrics allows cloud administrators to charge tenants based on utilization of the infrastructure. These include charges for power, CPU usage, bandwidth, and storage. Usage based metrics can have either simple or tiered pricing. Simple pricing is a flat rate regardless of volume usage. Tiered pricing is typically configured to either 1) charge for consumption exceeding a base amount included in the VM package or 2) to provide a discount for higher volumes of consumption.

OpenStack does not provide all of the metrics that a cloud provider might consider tracking and billing on. Some metrics must be derived by combining features and usage metrics into a new metric for billing. Here are three typical examples:

Instance vCPU hour – This metric is a measure of actual vCPU cycles consumed (vs allocated) by number of hours provisioned.
Instance RAM hour – Similar to Instance vCPU hour, this metric is a measure of the actual amount of RAM consumed (vs allocated) multiplied by the number of hours.

Disk MB hour – Rather than billing on allocated storage, some cloud providers would like to bill on the average or peak amount of storage consumed per hour.

Finally, users can augment the set of billable features by adding custom features directly into OpenBook. These user labeled features can include SLA bandwidth guarantees or additional services such as backup and recovery which carry additional rate plan charges or credits. Like the features pulled from OpenStack, user defined features are billed based on state and not usage since no metric is being tracked.

Summary

This report provides an overview of the components of a cloud bill and some of the options available to cloud administrators using OpenBook by Talligent. OpenBook has the flexibility to handle the rating and billing requirements for your OpenStack cloud regardless of whether you are offering IaaS, PaaS, DBaaS, or *aaS. We invite further discussion to understand your requirements and demonstrate OpenBook’s capabilities.