

# AWS Billing Engine

## 1. Billing Engine for Infrastructure as a Service (IaaS) platform

IaaS platforms like AWS, GCP, Azure etc. provide various infrastructure resources like computation resources, network resources, storage resources etc. For example, AWS provides EC2, VPC, elastic storage, S3 storage etc.

Customers will be charged based on various resources that they have used in a month. As part of this problem statement, we would like to emulate the functionality of the billing engine.

### Problem:

You will be given a set of records in a CSV file. Each record represents the usage of a resource by a customer for some time duration. You will have to implement a billing engine application, which generates monthly bills for all customers who have used one or more services.

### Input:

Following set of CSVs are provided as input to the billing engine.

**Note:** Here, the data are represented in tabular form just for clarity, but you should implement it, considering these data are provided in CSV file.

AWSResourceUsage.csv

Sr No	Customer ID	EC2 Instance ID	EC2 Instance Type	Used From	Used Until
1	CUST001	t3-wedjh	t3.medium	2021-08-15T10:00:00	2021-08-15T15:30:45
2	CUST002	t3-hsasa	t3.medium	2021-06-18T10:00:00	2021-08-15T15:30:45
3	CUST001	t3-gsadjh	t3.small	2021-08-05T10:50:00	2021-08-15T12:33:48
4	CUST001	t3-wedjh	t3.medium	2021-07-10T11:45:00	2021-07-15T15:30:45
5	CUST002	t3-jsakl	t3.small	2021-08-05T10:50:00	2021-08-15T12:33:48
6	CUST002	t3-hsajk	t3.micro	2021-07-10T11:45:00	2021-07-15T15:30:45
7	CUST001	t3-gsadjh	t3.small	2021-08-08T11:58:00	2021-08-08T12:35:00
8	CUST001	t3-hasgjh	t3.small	2021-08-08T16:10:10	2021-08-08T18:05:10

Customer.csv

Sr No	Customer ID	Customer Name
1	CUST-001	ABC Corporation
2	CUST-002	XYZ Corporation

AWSResourceTypes.csv

Sr No	Instance Type	Charge/Hour
1	t3.micro	\$0.0104
2	t3.small	\$0.0209
3	t3.medium	\$0.0418

### Output

After processing the input data, the billing engine should generate a monthly bill for each customer, i.e. 1 CSV file per customer per month.

For above example, the billing engine will generate 5 files as below. Each file represents a monthly bill for a customer.

1. CUST001\_JUL-2021.csv
2. CUST001\_AUG-2021.csv
3. CUST002\_JUN-2021.csv
4. CUST002\_JUL-2021.csv
5. CUST002\_AUG-2021.csv

CUST001\_AUG-2021.csv will look like as follows. Here again, the data are represented in tabular form for clarity, but your billing engine should generate appropriate csv file.

ABC Corporation					
Bill for month of August 2021					
Total Amount: \$0.3344					
Resource Type	Total Resources	Total Used Time (HH:mm:ss)	Total Billed Time (HH:mm:ss)	Rate (per hour)	Total Amount
t3.small	2	242:20:48	243:00:00	\$0.0209	\$0.0836
t3.medium	1	05:30:45	06:00:00	\$0.0418	\$0.2508

**Notes:**

1. Customer will be charged per hour, i.e. if customer has used an instance for 15 mins, the customer will have to pay for 1 hour for that usage duration.
2. Additionally, each different events will be considered as separate usage event for billing purpose. For example, if customer has used an EC2 instance for the following durations,
  - a. 2021-08-10T10:20:00 to 2021-08-10T10:40:00 (20 mins)
  - b. 2021-08-10T15:35:40 to 2021-08-10T15:45:40 (10 mins)

Here, customer has used an EC2 instance for 30 minutes in total, but during 2 separate timelines and so the customer will be charged for total 2 hours

3. AWSResourceUsage.csv will have usage data for all customers but may not be in a particular order
4. Similarly, all these records may not be a chronological order, e.g. Record 4 represents the event happened in the month of July, but record 1 represents the event happened in the month of August
5. Some of the events may span across months, e.g. Record 2 indicates that the customer has used an EC2 instance from 18-June to 15-August. However, while generating monthly bills, the total usage should be proportionally distributed across month of June, July and August
6. Total resources are computed based on different EC2 instance id, e.g. CUST001 has used two different instances ("t3-gsadjh" and "t3-hasgjh") of type t3.small in month of August and so, total resource count is 2 for instance type t3.small
7. Total Used time represents the accumulated time duration of the actual resource usage over entire month

## Enhancement 1:

### Provide support for multiple regions

AWS provides services in many different regions and continents, e.g. US, Europe, Asia Pacific etc and so,

1. A customer can provision instances within any of the supported regions, e.g. I can provision 2 instances of type t3.small in the US region and 3 instances of type t2.medium in Asia Pacific region.
2. For a given instance type (e.g. t3.small), the charge might vary based on the region, e.g. t3.micro is charged at \$0.0104/hr in US East (Ohio), but charged at \$0.0112/hr in Asia Pacific (Mumbai). You may refer [this](#) link to get more idea about the EC2 pricing in different region.
3. Additionally, while generating the bill, the customer would also like to segregate the itemized bill per region.

### Provide support for free-tier eligible EC2 instances

AWS provides free trial for t3.micro or t2.micro as per following rules,

1. 750 hours free/month for total t3.micro Linux instance
2. 750 hours free/month for total t3.micro Windows instance
3. A region will support either t3.micro or t2.micro as free tier eligible instance type, but not both

4. Free trial period is for 12 months, post that, regular charges would apply
5. Customer can utilize free hours in any combinations, e.g.
  - a. 1 t3.micro instance, running all the time for whole month, i.e.  $24 \times 30 = 720$  hours
  - b. 2 t3.micro instances, running continuously, but only for 15 days, i.e.  $2 \times 24 \times 15 = 720$  hours
  - c. 2 t3.micro instances, running 12 hours/day, but for whole month, i.e.  $2 \times 12 \times 30 = 720$  hours
6. Customer will get 750 hours/month free for Linux and Windows instances individually, i.e. collectively customer can utilize 750 hours of Linux + 750 hours of Windows = 1500 hours. However, if customer uses 600 hours of Windows and 850 hours of Linux, then even though total hours is 1450 (within limit of 1500), the customer has to pay nil for Windows instances, but will be charged for extra 100 hours of Linux usage
7. Free usage hours do not roll over to next month
8. Free usage hours are calculated across all regions, e.g.
  - a. If customer uses 1 t3.micro Linux instance for 24 hours x 10 days = 240 hours in "US East (Ohio)" region, and uses 1 t3.micro Linux instance for 24 hours x 25 days = 600 hours in Asia Pacific (Mumbai) region, then collective usage is  $240 + 600 = 840$  hours and so customer will be charged for additional 90 hours
  - b. Another scenario is, if customer uses
    - i. 1 t3.micro instance in US East (Ohio) for 24 hours x 10 days = 240 t3.micro hours in US East (Ohio)
    - ii. 1 t2.micro instance in US East (Ohio) for 24 hours x 15 days = 360 t2.micro hours in US East (Ohio)
    - iii. 1 t3.micro instance in Middle East region for 24 hours x 20 days = 480 t3.micro hours in Middle East region
    - iv. 1 t2.micro instance in Middle East region for 24 hours x 5 days = 120 t2.micro hours in Middle East region
    - v. Now, let's assume that the US East (Ohio) supports t3.micro as free tier eligible instance type, but Middle East region supports t2.micro as free tier eligible instance type
    - vi. So, in above case, the total eligible free hours = 240 t3.micro hours in US East (Ohio) + 120 t2.micro hours in Middle East region = 360 hours
9. Free tier eligible duration/months will be calculated as follows,
  - a. Let's assume that for a given AWSResourceUsage.csv
    - i. Total monthly bills generated for a customer are May-2019 to July-2021
      1. The free tier eligible months are May-2019 to Apr-2020 (1 year)
    - ii. Total monthly bills generated for another customer are May-2019, Sep-2019, Jan-2020, May-2020, Aug-2020
      1. The free tier eligible months are May-2019, Sep-2019, Jan-2020, because May-2020 and Aug-2020 falls outside of 1 year duration

### **Provide support for reserved instances**

In the above conversations so far, we are referring to instances, which are called OnDemand instances. AWS also supports Reserved instances as explained below.

Often, it happens that the customer knows in advance about the ec2 usage requirement. For example, customer wants to host their website in an ec2 instance and so that instance must be running for all the time, i.e. 24 hours x 365 days. For this use case, it is better to provision a reserved instance as explained below,

Let's say OnDemand pricing for t3.micro instance is \$0.125/hr and reserved instance price is \$0.100/hr. Considering this, if a customer uses OnDemand t3.micro, he will be charged 24 hours x 365 days x \$0.125/hr = \$1095, but if he chooses Reserved instance, then charges would be \$876.

However, if user has provisioned a reserved instance for a particular term (min 1 year, max 3 years), then user will be charged for that entire duration, e.g. 24 hours x 365 days (and irrespective of whether that instance was up or not during this term)

The customer will be billed monthly, i.e. if customer has reserved an instance for 1 year, then each month, the customer will have to pay 24 hrs x number of days in month x hourly rate (of reserved instance)

If customer provisions only 1 reserved instance in a region, e.g. 1 t3.micro in US East (Ohio) region for 1 year and if t3.micro is also eligible for free-tier pricing, the customer cannot take advantage of free-tier eligible hours, i.e. even though if customer's monthly usage is lesser than 750 t3.micro hours, the customer will have to pay as per reserved instance calculation explained above. However, if customer provisions one or more t3.micro instances (of type OnDemand) in same or different region, then those OnDemand type of instances will be eligible for free-tier discount explained above.

In order to fulfill above requirements, the input/output structure will be amended as follows,

AWSResourceUsage.csv is renamed to AWSOnDemandResourceUsage.csv with Region and OS as extra columns

Sr No	Customer ID	EC2 Instance ID	EC2 Instance Type	Used From	Used Until	Region	OS
1	CUST001	t3-wedjh	t3.medium	2021-08-15T10:00:00	2021-08-15T15:30:45	US (Ohio)	Linux
2	CUST002	t3-hsasa	t3.medium	2021-06-18T10:00:00	2021-08-15T15:30:45	Asia (Mumbai)	Windows

AWSReservedInstanceUsage.csv (new file with following structure)

Sr No	Customer ID	EC2 Instance ID	EC2 Instance Type	Start Date	End Date	Region	OS
1	CUST001	t3-hsgjh	t3.medium	2019-05-01	2020-04-30	US (Ohio)	Linux
2	CUST002	t3-hgsad	t3.medium	2020-02-01	2023-01-31	Asia (Mumbai)	Windows

Here,

- Start Date will be always be first day of some month
- End date will always be last day of some month
- Difference between End Date and Start date of a record must represent either 1 year or 3 years
- In AWSReservedInstaneUsage.csv, all records will have unique EC2 instance ID
- An instance will be of type OnDemand or Reserved, but not both, that is, if an EC2 instance ID is present in AWSReservedInstaneUsage.csv, the same will not appear in AWSOnDemandResourceUsage.csv and similarly if an instance id is present in AWSOnDemandResourceUsage.csv, the same will not appear in AWSReservedInstaneUsage.csv

Customer.csv file remains as is

Sr No	Customer ID	Customer Name
1	CUST-001	ABC Corporation
2	CUST-002	XYZ Corporation

AWSResourceTypes.csv will be enhanced as follows to support region and charge/hour for reserved instance

Sr No	Instance Type	Charge/Hour (OnDemand)	Charge/Hour (Reserved)	Region
1	t3.micro	\$0.0104	0.085	US (Ohio)
2	t3.micro	\$0.0125	0.0100	Asia (Mumbai)

Region.csv (new file)

Region	Free Tier Eligible
US (Ohio)	t3.micro
Asia (Mumbai)	t3.micro
Middle East	t2.micro

Monthly Bill file format (CUST001\_JAN-2021.csv)

ABC Corporation							
Bill for month of August 2021							
Total Amount: \$516							
Total Discount: \$400							
Actual Amount: \$116							
Region	Resource Type	Total Resources	Total Used Time (HH:mm:ss)	Total Billed Time (HH:mm:ss)	Total Amount	Discount	Actual Amount
US (Ohio)	t3.micro	2	780:50:45	781:00:00	\$500	\$400	\$100
US (Ohio)	t3.medium	1	05:30:45	06:00:00	\$5.50	\$0	\$5.50
Asia (Mumbai)	t2.medium	2	07:45:48	10:00:00	\$10.50	\$0	\$10.50

## Enhancement 2:

### Provide support for billing of Elastic IP service

AWS assigns a public IP to each EC2 instance, as long as it is in a running state. Once restarted, the public IP would change, which may not be acceptable if a customer uses an EC2 instance to host a website.

Alternatively, AWS provides Elastic IP (Reserved IP address), which is charged as follows,

1. If an Elastic IP is allocated and assigned to some running EC2 instance, there is no charge
2. If an Elastic IP is allocated, but not assigned to an EC2 instance, then it'll be charged at \$0.005/hr (charge may vary from region to region)
3. If an Elastic IP is allocated and assigned to an EC2 instance, but that instance is not running, then it'll be charged at \$0.005/hr for the duration during which associated EC2 instance was down
4. If a customer brings an IP address from his/her own pool (Bring your own IP address – BYOIP), then the customer will not be charged for these IP addresses. For example, if a customer has reserved/procured an IP address (12.35.78.108) from some other provider and plan to use it, then there is no charge for this IP address.

### Input:

#### ElasticIPAllocation.csv

Customer	Region	Elastic IP	Used From	Used Until	Your own IP?
CUST001	US (Ohio)	52.45.78.90	2021-05-10T15:45:50	2021-05-12T17:45:50	No
CUST001	IN (Mumbai)	52.45.88.91	2021-05-12T15:45:50	2021-05-18T17:45:50	Yes
CUST002	US (Ohio)	52.45.78.90	2021-05-20T15:45:50	2021-05-25T17:45:50	No
CUST001	US (Ohio)	52.45.78.90	2021-05-27T15:45:50	2021-05-28T17:45:50	No

**Note:** Same IP address can be reused across customers, e.g. once 52.45.78.90 was used by CUST0001 for the duration of 2021-05-10 to 2021-05-12 and then 2021-05-27 to 2021-05-28, whereas same IP address has been used by CUST002 for the period 2021-05-20 to 2021-05-25

#### ElasticIPAssociation.csv

IP Address	EC2 Instance	Associated From	Associated Until
52.45.78.90	t3-wedjh	2021-05-11T12:15:20	2021-05-11T18:15:20
52.45.78.90	t3-wedjh	2021-05-27T16:15:20	2021-05-28T10:15:20
52.45.88.91	t2-sadhg	2021-05-12T15:50:50	2021-05-18T17:40:50

#### ElasticIPRates.csv

Region	Rate/Hour
US (Ohio)	\$0.005
IN (Mumbai)	\$0.006



**Output**

Add itemized section in monthly bill as follows,

Region	IP Address	Total Allocation Time	Total Billed Time	Amount
US (Ohio)	52.45.78.90	50:45:30	20:00:00	\$0.010
In (Mumbai)	52.45.78.91	25:45:30	00:00:00	\$0.000

Total Monthly Amount should reflect accumulated charges incurred for using Elastic IP addresses.