

IBM Cloud Carbon Calculator

ENERGY & CARBON QUANTIFICATION METHODOLOGY

Disclosure:

Output that is provided by the carbon calculator is provided “as-is” for informational purposes only and is based on cloud services information that are provisioned by client in client’s IBM Cloud account. Output is provided in a format according to GHG protocol standards. The data sources and tools used to calculate client's emissions are subject to change by IBM without notice. Client is responsible for confirming accuracy of any calculator output for purposes of client’s compliance with any applicable regulatory obligations or for any other purpose.

Table of Contents

1. Introduction	4
2. Relevant Definitions and Terminology	5
Locations	5
Regions and data center locations	5
IBM Cloud services.....	5
Service Usage Units.....	7
3. Energy and GHG Emissions Calculation Method	4
Principle of Completeness	8
Allocation Methods.....	11
Allocating Electricity Consumption to Virtual Machines (VMs).....	13
Service Calculation Method.....	13
Apportioning Management Overhead (among Services and Customer Accounts).....	15
Accuracy	16
4. IBM Cloud Services	17
IBM Cloud Virtual Servers for Virtual Private Cloud (VPC).....	17
IBM Cloud Kubernetes Service (IKS) for VPC.....	18
IBM Cloud Object Storage (COS)	19
5. Obtaining the Carbon Emission Factors (CEF)	20
6. Calculating the GHG Emissions	21
7. Validation	22

IBM(R) Cloud Carbon Calculator - Methodology and Principles

1. Introduction

IBM Cloud is delivering enterprise transformation. From the most highly regulated industries to every business, IBM Cloud secures, integrates, hosts, and distributes enterprise workloads and data across hybrid cloud landscapes and digital ecosystems, within the enterprise and beyond.

IBM Cloud combines Platform as a Service (PaaS) with Infrastructure as a Service (IaaS) to provide an integrated experience with an advanced public cloud approach using Red Hat OpenShift. The platform scales support for both small development teams and organizations, as well as large enterprise businesses worldwide.

IBM Cloud plays a critical role towards achieving IBM's energy conservation, renewable electricity procurement, and greenhouse gas (GHG) emissions reduction targets, including IBM's goal to reach net zero GHG emissions by 2030. For more information about IBM's goals around environmental sustainability, please visit [here](#).

Like IBM, our clients care deeply about being environmentally responsible organizations and are increasingly interested in understanding the GHG emissions associated with their workloads running in IBM Cloud's environment. In support of this, IBM Cloud is advancing its long-used non-automated, report-based carbon estimation and is making it available to our clients through the IBM Cloud Carbon Calculator. Enabling them to get a detailed view of their cloud electricity consumption and associated GHG emissions.

When clients have visibility into this information, they can make informed decisions about management of their cloud consumption and understand how this affects their sustainability goals. IBM is committed to working with our clients to bring clarity to these decisions and the IBM Cloud Carbon Calculator is a key enabler to achieve that.

The purpose of this document is to bring transparency and credibility to this process by providing a clear description of the methodology used by the IBM Cloud Carbon Calculator to allocate electricity consumption and associated GHG emissions to IBM Cloud clients.

2. Relevant Definitions and Terminology

Locations

Location is defined as the geographical space where the service instance for the account is being deployed to. This shows the city name for the location. For display purposes, locations aggregate the information for the various datacenters inside the campus or buildings.

Note: The calculations are done by locations are used for display purposes and to reduce the complexity of the dashboard.

Regions and data center locations

IBM Cloud has a resilient global network of data centers to host our clients' workloads. These data centers are hosted in stand-alone facilities, in Single Campus Multizone Regions (MZR) and Multizone Regions (MZR). See below.

1. **Data centers:** In addition to selecting a region for a resource, users can select from a list of the IBM Cloud data centers. Data centers host compute, network, and storage resources and include energy consumption from supporting infrastructure (power generation and distribution, and cooling) used for services and apps.
2. **Regions**
 - a. **Single-campus MZR** contain three availability zones in different sections of the same building or within multiple buildings on the same campus where the power, cooling, networking, and physical security dependencies overlap, but are not identical between any two availability zones. This setup ensures a level of continuous availability and resiliency of any one system outage, planned or unplanned.
 - b. **Multizone regions (MZR)** are composed of three or more data centers (or zones) that are independent from each other to ensure that single failure events affect only a single zone. MZR provide low latency (< 2-milliseconds) and high bandwidth connectivity (> 1000 Gbps) across zones.

For more information, click [here](#).

IBM Cloud services

IBM Cloud services include options for compute, storage, networking, end-to-end developer solutions for app development, testing and deployment, security management services, traditional and open-source databases, and cloud-native services.

The IBM Cloud Carbon Calculator includes methodologies to estimate electricity consumption and associated GHG emissions for the following Cloud Services with coverage for additional services being added in future versions of the tool:

1. IBM Cloud® Virtual Private Cloud
 - a. IBM Cloud® Virtual Servers for Virtual Private Cloud
2. IBM Cloud Object Storage
3. IBM Cloud Kubernetes Service
4. Bare Metal (Classic)
5. Bare Metal (VPC)
6. Firewall Services
7. H/ware Security Module
8. H/W Load Balancer (Classic)
9. Block Storage Classic
10. Block Storage VPC
11. File Storage Classic
12. File Storage VPC
13. VPN for VPC
14. IBM Cloud DNS Services
15. Network Load Balancer
16. Application Load Balancer
17. SAP on Classic
18. SAP on VMware
19. D.bases Elasticsearch
20. D.bases for MongoDB
21. D.base for MySQL
22. D.base PostgreSQL
23. Databases for Redis
24. Msgs for RabbitMQ
25. D.bases for EtcD
26. D.bases for EDB
27. Code Engine
28. Event Streams
29. Event Notifications
30. App Configuration
31. Key Protect
32. Secrets Manager
33. VMware Solutions
- 34.

To learn more about IBM Cloud offerings, please visit [here](#).

Service Usage Units

Service Usage Units (Usage Units) are used to measure the “amount” or “volume” of any given service provided by IBM Cloud to our clients, and subsequently used to bill clients for that use. IBM Cloud clients may use a plurality of services in different regions, and IBM Cloud uses different Usage Units for specific resources and services.

For example, for IBM Cloud Kubernetes service, IBM Cloud meters a large number of Service Usage Units on which it bases billing. These Usage Units include things such as the number of virtual CPU hours (vCPU-hours), disk usage in gigabytes or gigabyte-hours and network usage in gigabytes.

Another example is Cloud Object Storage, where Usage Units include the total size of objects used in gigabytes, the bandwidth outgoing from the IBM Cloud for reads in gigabytes, and the number and type of operations.

The calculation method uses the Usage Units of individual IBM Cloud Service offerings that are most closely related to electricity consumption to allocate electricity consumption of shared resources to individual clients. For example, for the IBM Cloud Kubernetes service, the IBM Cloud Carbon Calculator bases its allocation of electricity consumption on the number of vCPU-hours consumed by the client.

Resource Groups

A resource group is a logical set of service instances that have been tagged by the client at deployment.

3. Energy and Carbon Footprint Calculation Method

The goal of the calculation method is to associate electricity consumption and associated GHG emissions allocation to the following:

1. Account,
2. Account and service used,
3. Account, service used, and location where the service is running (region or data center),
4. Account and resource group.

For all combinations listed above, our methodology provides electricity consumption and GHG emissions allocation with a month to month granularity.

Note: Every client account uses several services, and every service may run in multiple locations as depicted in the figure below. Thus, Item #1 above is the sum over Item #2, and Item #2 is the sum over Item #3. Item #4 is based on a logical distribution that the client has established for the account and therefore item #1 would also be the sum of item #4.

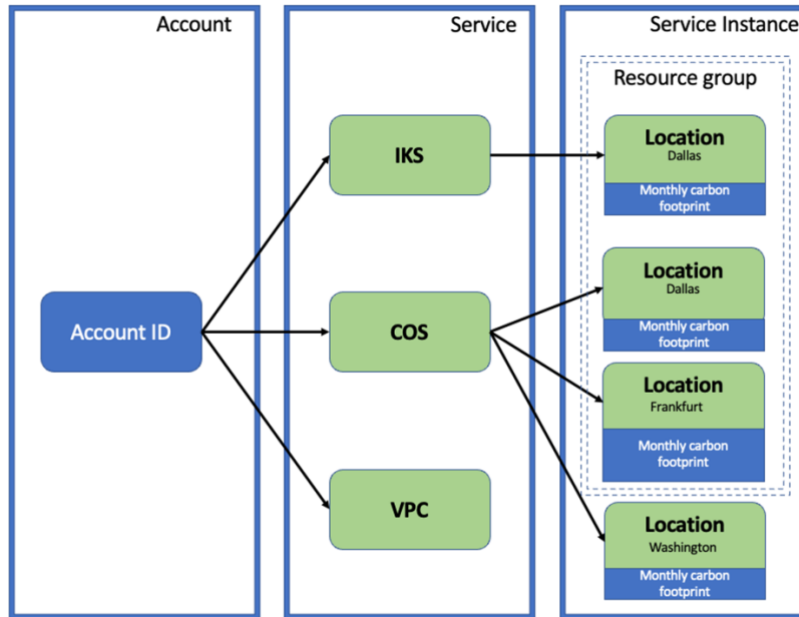


Figure 1. Illustration of a client account GHG emissions divided by services and locations, subsequently.

Concept of Completeness

This section describes the concept of completeness on which our IBM Cloud Carbon Calculator methodology is based on, and which is described as follows:

- The total electricity that a data center consumes must be completely allocated across all the client accounts that are using services running in this data center.
- In order to achieve this goal, we define the concept of a “Resource Element” and the concept of a “Provenance Chain” represented by a graph. See Below:
 - **Resource Element:**
 - A Resource Element is an instance of a resource. It can be a Physical Machine, a Virtual Machine, a Device (such as a router, or storage device), or a Cloud Service (e.g., IBM Kubernetes service). Resource Elements have in common that they are responsible for driving the consumption of electricity.

- Every Resource Element may be used by multiple users. A user can be a client account, or another Resource Element. For example, a Virtual Machine is a User of a Physical Machine.
 - Use relationships define how electricity is allocated across Resource Elements. If a Resource Element “A” uses a Resource Element “B”, then Resource Element “A” will be allocated a portion of the electricity consumed by Resource Element “B”.
 - Every Resource Element has its own defined electricity allocation method across its users. The allocation method is based on the Usage Unit(s) specific to the Resource Element type and may distinguish between Idle Power and Dynamic Power (defined under Allocation Methods below). It is important to understand that for every Resource Element, the electricity consumed by the Resource Element must be fully allocated to all its Users via the allocation method.
- **The Provenance Graph**
- The Provenance Graph is a directed graph where the nodes represent the Resource Elements, the arrows describe the use relationships between Resource Elements, and the labels represent the allocation of electricity consumption across Resource Elements. The sources¹ in the graph (i.e., a node without incoming arrows) are the physical devices for which electricity consumption can be either measured or estimated based on literature (or other methods as listed in Service Calculation Method below), if power metering is not present. The sinks (i.e., a node without outgoing arrows) are the client accounts.
 - The Completeness concept states that every node in the graph must satisfy that the sum of labels on all incoming arrows is equal to the sum of labels on all outgoing labels (i.e., the electricity consumed is allocated fully across the Users).

Note that, by design, the summation of all IBM Cloud clients’ electricity consumption is equal to IBM Cloud’s total electricity consumption globally.

- As shown in Figure 2. Example of Provenance Graph, every node in a graph for a Provenance Chain must satisfy that the sum of labels on all incoming arrows is equal to the sum of labels on all outgoing labels.

Examples:

Assume that an IBM Kubernetes Service runs in a data center, on two Physical Machines, and consumes a total amount of 50 watt-hours (Wh) in the month of January.

¹ The term sources here differs from the term source defined on the GHG Protocol. According to the GHG protocol a source is any physical unit or process which releases GHG into the atmosphere, and a sink is any physical unit or process that stores GHGs; (usually refers to forests and underground/deep sea reservoirs of CO2)

Further, let's assume that there are three (3) different User Accounts of this IBM Kubernetes Service (IKS). User Account 1 used a total of 5 vCPU-hours, the User Account 2 a total of 10 vCPU-hours, and User Account 3 a total of 10 vCPU-hours. Then the electricity consumption allocation for User Account 1 is 10 watt-hours, and User Accounts 2 and 3 each get allocated 20 watt-hours, in proportion with their usage, based on the IKS's Usage Unit specific to this service, vCPU-hour. In this example, a use relationship connects the IKS (which is an example of a Resource Element) to the Physical Machines, and the User Accounts to the IKS service.

How did we calculate the total electricity that the IBM Kubernetes Service (IKS) consumed? We looked at all the sources of electricity based on the use relationship. In this simple example, we assume that the IKS was running on two Physical Machines (PMs). The first Physical Machine consumed 10 watt-hours, and the second Physical Machine consumed 30 watt-hours. However, IKS is also a User of the Monitoring Service consuming 10 watt-hours (in a Microservice architecture it is common for Services to use other Services). Based on the above, the total of 50 watt-hours consumed by the IKS is divided into each User Account based on its own allocation method (e.g., vCPU-hour, bandwidth). The vCPU-hour is used for our example, making 20, 20, and 10 watt-hours allocated into User Accounts 1, 2, and 3, respectively (Figure 2).

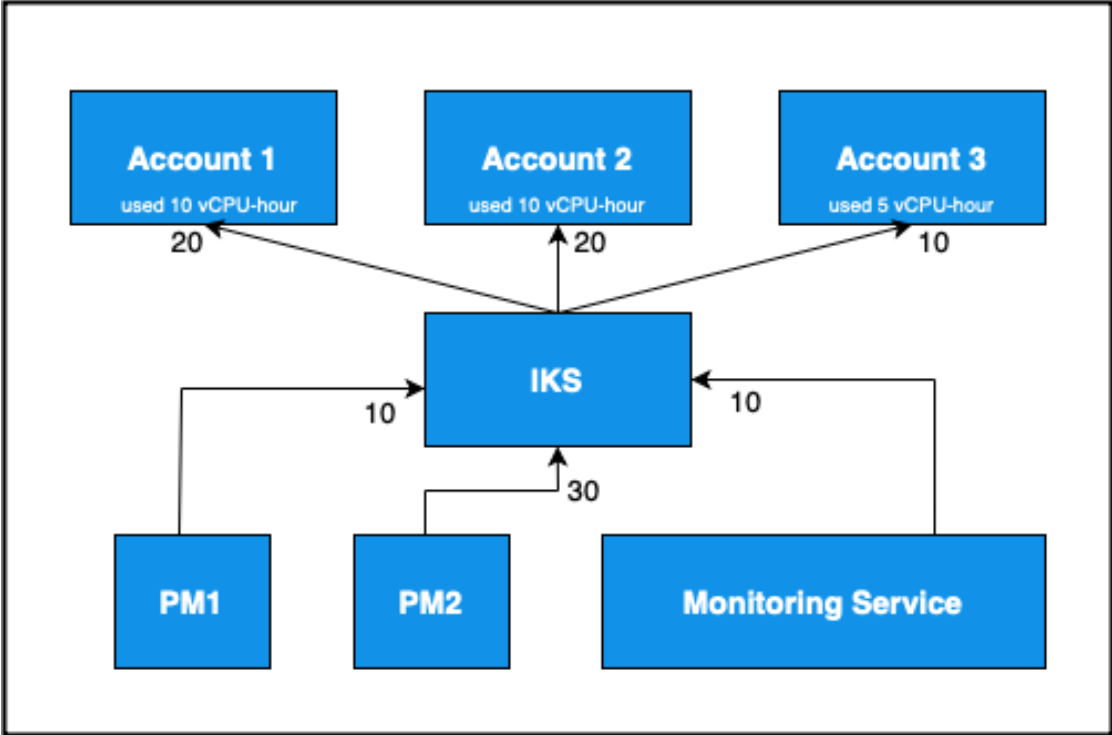


Figure 2. Example of Provenance Graph. Total energy consumption by IKS is 50 Wh. Accounts 1, 2 and 3 are allocated 20 Wh, 20 Wh, and 10 Wh, respectively. Direct arrows represent resource use relationship, and the labels on arrows represent the allocated energy (Wh).

Our methodology includes all electricity consumption associated with IBM Cloud's data center operations, including from IT and network equipment operated by IBM Cloud, and electricity

consumption from cooling and other support infrastructure operated by our co-location landlords where applicable and specifically assigned to IBM Cloud.

The following elements are out of scope of our calculations:

1. Embodied carbon emission of IT equipment hosted in IBM Cloud data centers (i.e., the energy and carbon footprint associated with the raw material and manufacturing of such equipment, including servers, racks, and networking equipment). This is because our approach computes only the operational or in-use carbon within a data centre.
2. The energy consumption associated with network, IBM Cloud Points-of-Presence (PoPs), or any other infrastructure physically located outside of IBM Cloud's data center facilities and hence outside of IBM's operational control, including client devices, external communication networks, etc.,
3. Any other value chain emissions associated with the products and services offered by IBM Cloud,
4. The energy consumption and emissions for transporting all the IT and non-IT equipment to run the services in the data centers,
5. The energy consumption and emissions associated with processing and disposing the IT and non-IT equipment at end of use (e.g., repair, refurbishment, remanufacturing, dismantling, landfill, recycle).

Allocation Methods

Different allocation methods have been developed for every Resource Element (R) based on their specific Usage Units. Any given Resource Element "R" may be used by "n" number of Resource Elements r_1, \dots, r_n (r_1 through r_n are therefore Users). If R consumes a total amount of electricity E, then this amount of electricity consumption must be allocated across its Users r_1 through r_n based on quantity of each usage unit that each user consumes. Resource Element R has its own usage units, $u_{R,1}$ through $u_{R,m}$. Typically, we expect to have one or two usage units per Resource Element, but the methodology works with any number m of usage units. More specifically, the total electricity E consumed by R is first divided among R's usage units as $E_{R,1}$ through $E_{R,m}$. The electricity attributed to each usage unit of R (e.g., $E_{R,i}$ of the i^{th} usage unit) is then divided among the users in proportion to the quantity of that usage unit consumed by each user.

In addition, the methodology distinguishes between the resource element's idle power consumption, E_{idle} , and its dynamic power consumption, E_{dyn} , where $E = E_{idle} + E_{dyn}$.

Idle power is the power consumed or attributed to a Resource Element when it is in idle state, that is, when it is not utilized to perform any work. *Dynamic power*, also referred to as *active*

power, is the power consumed or attributed while the Resource Element is utilized to perform work.

Weight functions are defined to allocate electricity consumption across the Users of a Resource Element R based on its User's usage units. Two weight functions are defined, one to allocate idle power consumption E_{idle} and another one to allocate dynamic power consumption E_{dyn} to the Users (r_1, \dots, r_n).

Allocation of idle and dynamic power to the Users is also based on the usage units. Usage Units are associated with different weight factors based on their specific contributions toward total electricity consumption of a Service. Note that these weight factors are constants that are determined via experimentation and statistical methods, such as linear regression, and may change over time.

For the sake of simplicity, we assume here a linear relationship, but this may not always be the case. The actual form of the mathematical function which is used to split idle and dynamic power across tenants based on usage data is determined via experimentations and AI learning techniques.

In its most general form, the allocation function is as follows:

Let $u_i(A, S, L)$ be the Usage Unit i (e.g., vCPU-hour) for client account A , using a service S , in a location L during a specified period of time. Let $\sum u_i$ be the total amount of usage units (i.e., total amount of vCPU-hour) used by all client accounts using the service S in the location L during the specified period of time. Thus, sum of u_i over all accounts can be described as $u_i(S, L)$. Further, let E_{idle} be the total idle power consumed by the service S in location L , and E_{dyn} the dynamic electricity consumption. Thus, $E = E_{idle} + E_{dyn}$, where E is the total electricity consumption of service S in location L .

Assuming a linear relationship, Let a_i be the weight factor used to allocate E_{idle} based on the Usage Unit's relative amount across all client accounts, and let b_i be the weight factor used to allocate E_{dyn} based on Usage Unit's relative amount across all client accounts. Then, the mathematical expression to calculate the electricity consumed by client account A on service S at location L , $E(A, S, L)$, is given below:

$$E(A, S, L) = E_{idle} \times \sum \frac{a_i \cdot u_i(A, S, L)}{u_i(S, L)} + E_{dyn} \times \sum \frac{b_i \cdot u_i(A, S, L)}{u_i(S, L)}$$

Allocating Electricity Consumption to Virtual Machines (VMs)

One application of this methodology is to allocate the electricity consumed by a Physical Machine (PM) to the Virtual Machines (VMs) running on it. Multiple VMs can run on a single PM.

According to the [ICT Sector Guidance of the GHG Protocol Product Life Cycle Accounting and Reporting Standard](#), the size of the VMs (such as the number of virtual cores allocated to it) should be used (as weights) to allocate the idle power consumption of the PM, while degree of utilization of the VMs, factored by size, should be used to allocate the dynamic power consumption of the PM. Therefore, the relevant usage units for this application are:

1. Size (such as the number of virtual cores used by the VM), and
2. Utilization (such as average CPU utilization multiplied by the number of virtual cores), where, one is used primarily to split the idle power of the machine, and the other one is only factored in the fairly split the dynamic power of the machine.

Service Calculation Method

We calculate the electricity consumption and corresponding GHG emissions in three basic steps outlined in Figure 3.

1. The **first step** estimates the IT electricity consumption for a service in a every IBM Cloud location. We meter electricity consumption for every physical machine (PM), and any device, such as storage device. We then allocate electricity consumption to the individual virtual machines (VM) running on a every PM (as described above). The output of this step is the IT electricity consumption allocation for every physical machine (PM) and virtual machine (VM). In case power metering is not available for a given PM, we estimate the electricity consumption for it by using other techniques, e.g., using available literature, power specification of a server and its components obtained from hardware manufacturer specifications, or running benchmarks on sample physical machines. In some cases, wherein an entire rack is dedicated to a specific service, aggregate power consumed by all the servers in a rack, obtained by sensing rack power supplies, is used.

Based on this information, we can now calculate the total power consumption for every service in every location. If a service S, in a location L runs on a set of physical or virtual machines, and use other devices, the summation over the physical machines (if used) or virtual machines (if used), and storage and network devices give us the total electricity consumption for the service in the location.

2. In the **second step**, we understand the total electricity consumption incurred due to IT devices for every service (S) and location (L) - (In this step, the electricity consumption by non-IT devices in the data center, such as HVAC equipment is not considered. Other sources of energy consumed by non-IT devices are also not included here). Then, we allocate electricity consumption to every client account for each specific service being used by the client in each specific location. This is done based on the amount of Service Usage Units being used by the client account relative to the total amount of Service Usage Units used and following the template described above.

The calculation method is service specific and is based on the Service Usage Units allocated to the client account. If a service only has one Service Usage Unit, e.g., the unit vCPU-hour, then the allocation of IT electricity consumption to a particular client account will correspond to the ratio of the service usage units (vCPU-hour) for that account relative to the total usage of all client accounts in this location (i.e., the total number of Service Usage Units). If multiple types of Service Usage Units are used (e.g., volume and bandwidth) then the ratio of each Service Usage Unit type is fed into a weight function.

Where it makes sense (e.g.,), Idle Energy is handled differently (i.e., different weights) than Dynamic energy. For a service, Idle Energy is the Energy consumption when no workloads run, i.e., it includes the Idle power of all PMs and the Management overhead.

Management overhead consists of the electricity consumption by all the ancillary hardware equipment and software components that are required for supporting a service. Examples include Cloud control services such as metering and security services, network services for inter-server, inter-rack and internet connectivity, service specific overhead such as those of master nodes used by IKS.

The output of this second step is the allocation of IT electricity consumption to every Service, Location and Account.

3. Finally, in the **third step**, we factor in the Power Usage Effectiveness (PUE) to account for electricity consumption associated with supporting infrastructure, such as cooling and other energy services provided to the locations to arrive at total electricity consumption per service, location, and client account. We then proceed to multiply this total electricity consumption by the grid-specific Carbon Emission Factor (CEF) applicable for the location. PUE and CEF are specific to each location.

These steps are described in further detail in this figure.

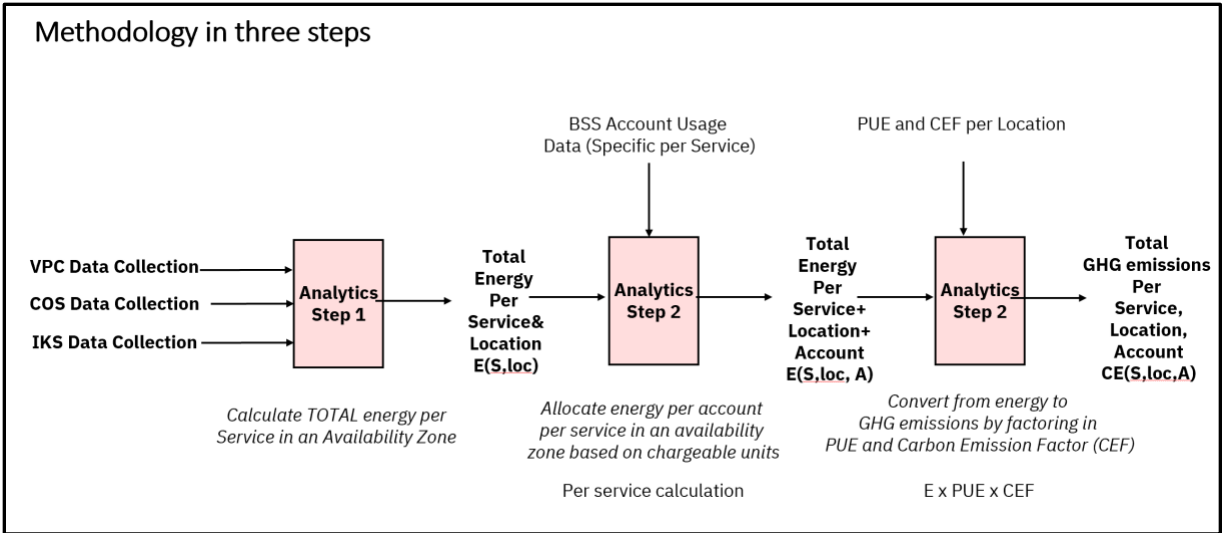


Figure 3. Flow diagram of the methodology

Figure 3. Flow diagram of the methodology and Figure 4 further illustrate how the Service Calculation Method works. Physical machine energy allocation is determined first, then virtual machine energy allocation, then service at a location energy allocation, then splitting across accounts occur, and last, PUE and CEF are factored in. One detail that is not illustrated is to factor in the service provenance chain. If a service S1 is using a service S2 and IBM Kubernetes Service is using other VPC services (e.g., network, load balancer services), then based on the portion of usage of S2, a fraction of the total energy that S2 consumes will be allocated to service S1. Please see Figure 2.

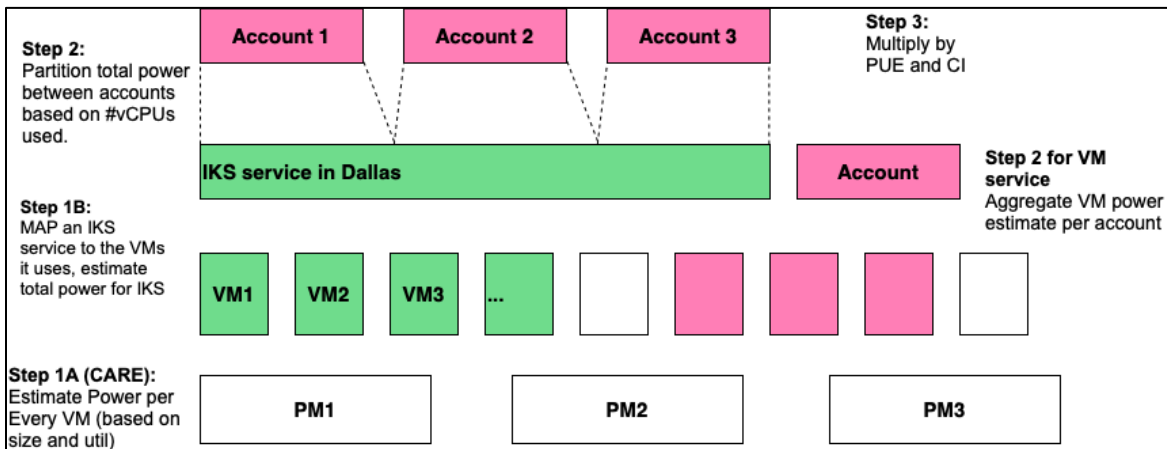


Figure 4. Illustration of Service Calculation Methodology

Apportioning Management Overhead (among Services and Customer Accounts)

When two or more services share a management component, such as the metering service or backbone networking equipment, we divide the energy consumed by the component among the services in proportion to the number of accounts that use each service. The share of energy allocated to each service is in turn apportioned among the customer accounts of the service in proportion to their service usage units (similarly to how the primary energy consumed by the service is apportioned).

Accuracy

Aligning with [IBMs Position on Scope 3 GHG emissions](#) document, we define Accuracy as a measure of the degree of correctness with which a measurement reflects the true value of the parameter being assessed. Accuracy is also listed as one of required principles to be adhered to in GHG accounting and reporting by the GHG protocol in [Product Life Cycle Accounting and Reporting Standard](#). For GHG emissions calculations, we consider several factors that can be directly measured and that are being included as primary data in our methodology in an effort to increase the level of accuracy and precision in the estimation of electricity consumption and associated GHG emissions.

Data uncertainty: There is a certain degree of uncertainty introduced into our calculations based on estimations, to ensure that the quantification of GHG emissions is systematically never over nor under actual emissions and that uncertainties are reduced as much as possible:

- Currently we are using the primary or most relevant service resource unit to allocate the power consumption,
 - Since any given service may have more than one utilization unit, as we introduce additional utilization units, the allocation becomes more accurate.
Reconciling the allocation with the location total power consumption data (see Concept of completeness section above), this monthly reconciliation ensures we are not over or under-reporting GHG emissions and that all of the power is allocated to either IBM or the respective accounts as per the allocation methodology. It also provides a means to identify and reduce uncertainties over time. As it allows us to track progress towards making this discrepancy lower in time.

Limited Monitoring capability: The variations in the non-primary resource units may have an impact in the power consumption, but they cannot be currently allocated as we do not have that level of precision in our power/resource monitoring yet.

Calculating the GHG metrics entails deploying additional infrastructure and consuming additional processing cycles for the purpose, resulting in associated energy consumption and GHG emissions. The higher the accuracy, the greater the emissions are expected to be. Hence, there is a need to strike a balance between the accuracy of computation of metrics and GHG emissions. As the efficiency of computing the metrics improves, accuracy obtained for a given quantum of GHG emissions is also expected to improve.

Reporting accuracy: With the current methodology, as we gather more data, our estimations will become more accurate. The introduction of new hardware and software into our cloud backbone allows more precision on direct measurement of power consumption, consequently, we will be able to provide more accurate data over time.

Although, the methodology does not change, we are able to introduce better data collection, the resulting numbers will be less estimation and more precise over time without generating a large overhead on the services or our power consumption and associated GHG emission.

6. IBM Cloud Services

In this section we lay out the methodology that we use for calculating the GHG emissions for infrastructure based on Intel x86 servers which form the majority of IBMs datacenter footprint. Although the basic approach will remain the same, we anticipate that there will be some changes needed for IBM's Power™ servers and Z-systems™ servers. We will cover these platforms in a future update to this document. This document will lay out the Carbon Calculator approach for three initial exemplar IBM Cloud services, but it is extensible to the entire catalog. These three services are IBM Cloud Virtual Servers (VM) for VPC, IBM Kubernetes Service (IKS), and IBM Cloud Object Storage (COS).

IBM Cloud Virtual Servers for Virtual Private Cloud (VPC)

IBM Cloud Virtual Server for VPC offers IBM Cloud clients fast-provisioning compute capacity with the highest networking speeds and most secure, software-defined networking resources available on the IBM Cloud. Users of Virtual Server or Virtual Server Instance (VSI) service will start/stop and use Virtual Machines (VMs) managed by the VPC infrastructure. The virtual server usage (e.g., timing of using certain profile of VM server) is logged into the VPC management system. For example, a profile of a virtual server is 2 vCPU x 4GB Memory (2x4) and the virtual CPU usage is *vCPUh*. For every datacenter location *L*, we first calculate the total electricity $E(VSI, L)$ consumed by all VPC VSI in that data center in watt-hours. We then use the proportionality of the vCPU usage to calculate a user's electricity consumption. The electricity consumption for running a client VSI in a datacenter becomes as follows:

$$E(A, VSI, L) = \frac{vCPUh(A, VSI, L)}{vCPUh(VSI, L)} \times E(VSI, L) + E(A, VSI, L, mgmt)$$

where A , VSI and L indicate the account, virtual server instances service and datacenter location, respectively. Since the CPU usage is the most important metric for the power/energy consumption, the energy proportionality of the vCPU usage for every hour in a datacenter is accounted for a customer's electricity consumption. The model can be extended and become more accurate by including additional service usage units related to hardware utilization (e.g., memory hours, I/O hours) as described in the second step. We factor in not only the electricity consumption of physical servers that are running client VMs, but also the servers that are used to manage the VMs. The management overhead for VSI and their electricity consumption will be proportionally divided by the size of VSIs and add to the equation above, $E(A, VSI, L, mgmt)$.

VSI with GPUs

Some of the VSI offerings come with Graphics Processing Units (GPUs). GPUs are also one of the major consumers of electricity in servers, and hence GPU usage needs to be accounted for in per-account energy computations. Our approach for VSIs with GPUs in VPC environment is as follows. For every datacenter location L , we first calculate the total electricity $E(VSI_g, L)$ consumed by all VPC GPU VSIs in that data center in watt-hours. We then use the Thermal Designed Power (TDP) of CPUs and GPUs to determine the ratio in which the total electricity consumption of the service is to be split into two components, across CPU's and GPU's. Finally, each account's share of each component is obtained using the fraction of the account's total usage of each component. Thus, the electricity consumption attributed to a client GPU VSI in a datacenter is given by:

$$E(A, VSI_g, L) = E(VSI_g, L) \times \left(w \frac{vCPUh(A, VSI_g, L)}{vCPUh(VSI_g, L)} + (1 - w) \frac{vGPUh(A, VSI_g, L)}{vGPUh(VSI_g, L)} \right) + E(A, VSI_g, L, mgmt)$$

where A , VSI_g and L indicate the account, GPU virtual server instances service and datacenter location, respectively. Since the CPU and GPU usages are the most important metric for the power/energy consumption, the energy proportionality of the CPU and GPU usage based on the TDP is accounted for a customer's electricity consumption. w indicates the ratio between the total TDP over all the CPUs and GPUs over all VPC GPU VSI servers in the data center. This model can be extended and become more accurate by including additional service usage units related to hardware utilization (e.g., memory hours, I/O hours) as described in the second step. We factor in not only the electricity consumption of physical servers that are running client VMs, but also the servers that are used to manage the VMs. The management overhead for VSI and their electricity consumption will be proportionally divided by the size of VSIs and add to the equation above, $E(A, VSI_g, L, mgmt)$.

IBM Cloud Kubernetes Service (IKS) for VPC

IBM Cloud Kubernetes Service is a managed offering that lets IBM Cloud clients create their own Kubernetes cluster of compute hosts to deploy and manage containerized apps on IBM Cloud. Similarly, to VSI, users of IBM IKS will also start and stop, and use a Kubernetes Cluster managed by the VPC infrastructure. The Kubernetes cluster usage (e.g., timing of using certain profile of IKS nodes and their usage) are logged by the VPC management system where the Kubernetes nodes are running in the VMs. For example, a profile of Kubernetes VM server would be 2 vCPU x 4GB Memory (2x4) and its virtual CPU usage is $vCPUh$. For every location L , we first calculate the total electricity $E(IKS, L)$ consumed by the IKS in that location in every hour. The electricity consumption of a running IKS instance is as follows:

$$E(A, IKS, L) = \frac{vCPUh(A, IKS, L)}{vCPUh(IKS, L)} \times E(IKS, L) + E(A, IKS, L, mgmt)$$

Where, A , IKS and L indicate the account, IBM Kubernetes Service, and datacenter location, respectively. The $vCPUh$ indicates the virtual CPU hours. The energy proportionality of the vCPU usage for a given time in a datacenter is accounted for a customer's electricity consumption for IKS as well. The model will be extended and become more accurate by including more service usage unit types related to hardware utilization (e.g., memory hours, I/O hours). The management overhead for IKS and their electricity consumption will be proportionally divided by the size of worker nodes and added to the equation above, $E(A, IKS, L, mgmt)$.

IBM Cloud Object Storage (COS)

Total energy consumed by COS service at a location: IBM's Cloud Object Storage (COS) provides flexible, cost-effective, and scalable cloud storage for unstructured data. The servers and storage devices providing this service are housed in racks of their own along with the top of the rack network devices. Hence, the total energy consumption for this service at a location can be obtained by aggregating the average power over time consumed by all the racks pertaining to this service. In locations where rack power is not obtained by direct measurements, the total energy is calculated by sensing the power of individual hardware devices, namely, accessor servers, slicestors (storage servers), other application servers, load balancers, and network switches. While the power consumption at the first three classes of devices above is directly sensed, that at the load balancers and network switches is estimated from their power rating available in hardware specification provided by the vendors. The estimation is performed as follows: While regular power measurements are not possible at the rack level, rack level power readings can be taken occasionally. We use a few such readings along with readings of individual accessors, slicestors and applications servers to determine the aggregate power consumption of load balancers and network switches. We

then distribute the aggregate power across individual load balancers and network switches in proportion to the thermal design power (TDP) rating provided by the device manufacturers.

Energy per account due to COS service at a location: The aggregate COS service energy determined using the method above can be apportioned among the accounts using the service in proportion to their usage of resources. The significant resources consumed by this service are disk storage, for storing, and network bandwidth (both external and internal), for accessing the objects in storage. The external bandwidth refers to the networking equipment and bandwidth of a location needed to establish communication with and transfer objects to the external world. Compute processing incurred for maintaining the storage objects and serving them is expected to be in proportion to the storage size and bandwidth and hence need not be explicitly and separately accounted for. Thus, energy per account for COS is computed using the following formula.

$$E(A, COS, L) = w_1 \times \frac{S(A_L)}{S(L)} \times E(COS, L) + w_2 \times \frac{B(A_L)}{B(L)} \times E(COS, L) + E(A, COS, L, mgmt),$$

Where $S(A_L)$ is the storage used by account A at location L, $S(L)$ is the total storage used by all the accounts at L, $B(A_L)$ is the bandwidth used by account A, and $B(L)$ is the total bandwidth consumed by all the accounts. The weighting factors w_1 and w_2 can be estimated from the energy proportion of storage and network devices. As described in Sec. 2 (Allocation Methods), these weighting factors are constants, determined via experimentation such as linear regression. The management overhead for COS and their electricity consumption will be proportionally divided by the bandwidth and storage usage of COS and add to the equation above, $E(A, COS, L, mgmt)$.

7. Obtaining the Carbon Emission Factors (CEF)

We follow the Greenhouse Gas [Protocol Scope 2](#) Guidance to calculate market-based emissions associated with the electricity consumption calculated in previous steps. Because emission factors typically are not updated more frequently than yearly by the respective issuing authorities, we use the most up-to-date data available for a period as long as one year. In calculating market-based emissions, we account for IBM Cloud’s consumption of renewable electricity at each location.

We calculate the GHG emissions using emission factors specific to the local electric grid where each location is, from the following sources:

Geography	Description
USA	Emissions and Generation Resource Integrated Database (eGRID), which is a comprehensive source of data from EPA on the environmental characteristics of all

	electric power generation in USA. For more information, see eGRID Subregion Total Output Emission Rates .
Canada	Canada National Inventory Report provided by United Nations Framework Convention on Climate Change (UNFCCC). For more information, see National Inventory Report (NIR) .
EU	European Residual Mix provided by the European Energy Certification System (EECS) of Association of Issuing Bodies (AIB). For more information, see European Residual Mix . https://www.eea.europa.eu/jms/greenhouse-gas-emission-intensity-of-1
UK	Conversion Factors for Greenhouse Gas (GHG) reporting by companies provided by the government of UK. For more information, see Greenhouse gas reporting .
Australia	Australian National Greenhouse accounts provided by Department of Industry, Science and Resources. For more information, see National Greenhouse Accounts Factors .
Japan	https://www.climate-transparency.org/wp-content/uploads/2021/10/CT2021Japan.pdf#page=6
Rest of the world	IGES list of Grid Emission Factors: https://www.iges.or.jp/en/pub/list-grid-emission-factor/en Will be eventually replaced by IEA emission factors.

Accounting for overhead data center electricity consumption

IBM Cloud data centers are, for the most part, hosted in third-party co-location facilities which results in IBM not owning or having operational control over facility infrastructure management, including non-IT services to our data center space such as cooling and lighting. To account for this additional non-IT electricity consumption, we use the metric of Power Usage Effectiveness (PUE). We obtain this data from our landlords at these facilities.

PUE is given by the ratio of the total electricity consumed by the data center (aggregate consumption by the IT and support overhead infrastructure) to that consumed by the IT infrastructure. Following the standard practice, we apportion the overhead electricity consumption among the clients in proportion to their IT electricity usage. For this purpose, we multiply each account’s IT electricity usage by the PUE factor.

8. Calculating the GHG Emissions

This section gives details about calculating the GHG Emissions for a client account, service, and location. For a client account, the GHG emissions associated with a service (S) use at a particular location (L) is given by:

$$GHG(A, S, L) = E(A, S, L) \times PUE(L) \times CEF(L)$$

The methodology calculates market-based emissions, following the GHG Protocol’s Scope 2 Guidance.

- Market-based emissions, $CEF(L)$ equals zero if the location is using 100% renewable electricity and, if that is not the case, $CEF(L)$ is equal to a supplier-specific emission factor, a residual-mix emission factor, or the local grid emission factor (in that order depending on what factors are available) from the sources listed above.

As explained above, all other accounting elements that we make available to clients, are aggregation over this basic entity such as $GHG(A,*,L)$ which is the GHG emissions for a client account at a particular location per service, or $GHG(A)$, which is the GHG emissions total, over all services in all locations for one account.

7. Validation

We perform a first-order validation of the accounting approach at each location by ensuring that the sum of the emissions assigned to all the customer accounts at the location is equal to the location's aggregate emission as computed using the aggregate electricity consumption of the location's data centers.

Disclaimers

© Copyright IBM Corporation 2023

IBM Corporation
New Orchard Road
Armonk, NY 10504

Produced in the United States of America
December 2023

IBM, the IBM logo, ibm.com, and IBM Cloud are trademarks of International Business Machines Corp., registered in many jurisdictions worldwide. Other product and service names might be trademarks of IBM or other companies. A current list of IBM trademarks is available on the Web at “Copyright and trademark information” at www.ibm.com/legal/copytrade.shtml.

This document is current as of the initial date of publication and may be changed by IBM at any time. Not all offerings are available in every country in which IBM operates. THE INFORMATION IN THIS DOCUMENT IS PROVIDED “AS IS” WITHOUT ANY WARRANTY, EXPRESS OR IMPLIED, INCLUDING WITHOUT ANY WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND ANY WARRANTY OR CONDITION OF NON-INFRINGEMENT. IBM will not be liable for any direct or indirect damages, including without limitation, lost profits, lost savings, or any incidental, special, or other economic consequential damages, even if it has been advised of the possibility of such damages based on the information in this document. IBM products are warranted according to the terms and conditions of the agreements under which they are provided.

Clients are responsible for ensuring their own compliance with various applicable laws and regulations. Clients are solely responsible for obtaining professional legal advice as to the identification and interpretation of any relevant laws and regulations that may affect the clients’ business and any actions the clients may need to take to comply with such laws and regulations.

IBM does not provide legal, accounting or auditing advice. IBM also does not represent or warrant that its services or products will ensure that clients are compliant with any applicable laws or regulations.