Estimating the Cost
of a GIS in the Amazon™ Cloud

Estimating the Cost of a GIS in the Amazon Cloud

An Esri White Paper

Contents	Page

Contents   **Page**

Estimating the Cost of a GIS in the Amazon Cloud

Introduction

Geographic information system (GIS) and information technology (IT) managers see the cloud as a compelling argument for leveraging off-premises hardware resources and reducing local infrastructure costs. But how does one go about calculating the cost of using cloud resources in lieu of using on-site infrastructure?

The purpose of this paper is to help the reader understand how costs are calculated for cloud services—using Amazon™ Elastic Compute Cloud (EC2) as an example—and how to estimate the costs of using Amazon cloud services with an ArcGIS Server cloud implementation: ArcGIS® Server on Amazon EC2.

What Is ArcGIS Server on Amazon EC2?

ArcGIS Server is the core server GIS software made by Esri, used for creating, managing, and distributing GIS Web services, applications, and data. ArcGIS Server is typically deployed on-premises within the organization's service-oriented architecture (SOA) or off-premises in cloud computing environments or data centers.

Esri offers ArcGIS Server on Amazon EC2 as a preconfigured solution for organizations wishing to deploy their GIS servers on established cloud infrastructures rather than use their own local infrastructures. ArcGIS Server on Amazon EC2 provides two Amazon Machine Images (AMIs): ArcGIS Server and an enterprise geodatabase configured on PostgreSQL.

An AMI represents a machine image or the state of a machine once it's completely configured for a certain need. In the case of ArcGIS Server on Amazon EC2, the ArcGIS Server AMI is used to launch instances of ArcGIS Server. Each instance is configured to run on Windows® 2008 and includes a 100 GB drive for GIS data. However, additional configuration options are available. These options are covered in more detail later.

Choosing Services Offered on Amazon

Amazon offers optional services that can be associated with an instance or group of instances. The pricing for these services is based on a pay-as-you-go pricing model.

Amazon Elastic Block Store

Elastic Block Store (EBS) is the storage that can be attached directly to an instance to become a local drive to the machine. The EBS can be detached and managed separately as needed. When an instance is launched from the AMI, it automatically includes attached storage that is ready to use for any custom data. For example, the ArcGIS Server on Amazon EC2 AMI and the enterprise geodatabase AMI each include 100 GB of EBS. The EBS has a life cycle that is separate from the instance so that if the instance is terminated, the EBS volume continues to exist. This allows users to retain and manage their data separately from the instance. A typical use of EBS is to create a snapshot or backup copy of project data. For more about Elastic Block Store, visit aws.amazon.com/ebs/.

Elastic IP Addresses

Elastic IP addresses provide an IP address that never changes inside the cloud. Typically, when a new instance is launched, it is assigned a temporary external IP address. However, each time the instance is terminated or replaced, this external IP address will change. In some cases, it is necessary to assign an IP address that remains static. The elastic IP address is associated with an account and not a particular instance. It can be maintained until the user chooses to explicitly release it. For example, an elastic IP address can be assigned to a user's Amazon Web Services (AWS®) account, then mapped to a particular ArcGIS Server instance when ArcGIS Server on Amazon EC2 is used. The IP address can be remapped to a replacement ArcGIS Server instance.

Elastic Load Balancing

Elastic Load Balancing automatically distributes incoming application traffic across multiple Amazon EC2 instances. It enables users to achieve even greater fault tolerance in their applications and provides the amount of load balancing capacity needed in response to incoming application traffic. Elastic Load Balancing can be associated with multiple instances within a single availability zone or across multiple zones to maintain application performance or to compensate for instances that stop working properly. For example, Elastic Load Balancing can be associated with a set of ArcGIS Server instances that support high-traffic or mission-critical applications. For more information about Elastic Load Balancing, visit aws.amazon.com/elasticloadbalancing/.

Auto Scaling

Auto Scaling ensures that the number of Amazon EC2 instances being used scales up (increases) during demand spikes to maintain performance, then automatically scales down during traffic lulls to minimize costs. Auto Scaling is enabled by Amazon CloudWatch and available at no additional charge beyond Amazon CloudWatch fees. For more information about Auto Scaling, visit aws.amazon.com/autoscaling/.

Amazon CloudWatch

Amazon CloudWatch provides visibility into AWS cloud resource utilization, operational performance, and overall demand patterns including metrics such as CPU utilization, disk reads and writes, and network traffic. For example, users can monitor use of their ArcGIS Server on Amazon EC2 instances in real time via the AWS Management Console, including EBS volumes and Elastic Load Balancing. For more information about Amazon CloudWatch, visit aws.amazon.com/cloudwatch/.

How AWS Data Center Costs Are Calculated when Running ArcGIS Server on Amazon EC2

First, the general concepts that contribute to the cost of cloud computing must be taken into consideration.

Data Transfer and Hourly Rates

The costs of running an instance on hosted infrastructure depend primarily on two factors—the hourly rate of the instance type and data transfer to and from the instance.

In general, if the data collection is small (e.g., no larger than 10 GB) and the expected network traffic to/from the instance is low (e.g., only 2 to 10 users are accessing the services a few times a day), then the majority of the cost will be generated by the hourly rate for the instance type.

However, as data size and network traffic to and from the instance increases, the relationship between hourly rates and data transfer rates changes. With large datasets and high traffic, the majority of costs are generated by the data transfer rate, not the hourly rate of the instance.
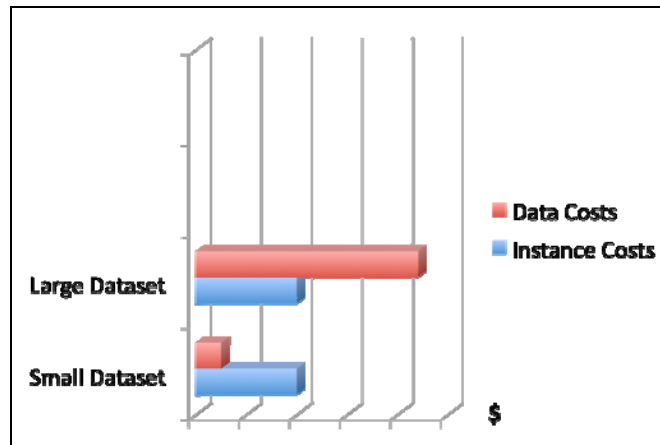
The Cost of Underutilized Resources On Amazon EC2, users can realize cost savings by making sure to use only the resources that are needed, when they are needed. For example, users can manage cloud resources by planning to use a smaller machine during times of low traffic, then add more or larger machines when traffic reaches a certain threshold. Starting machines and letting them run even when they're not in use or when they're not fully utilized will not allow an organization to fully realize cost savings in the cloud.

The Cost of Time When estimating cloud computing costs, time savings is often overlooked as a factor of potential costs. Typical time-saving variables include time to set up and maintain cloud resources.

Time to Set Up In traditional cases of implementing on-premises hardware, considerable time is spent going through the different steps of getting a machine ready to use. These steps include procurement of hardware and software, installation and configuration of the operating system and software, and network configuration.

In the case of using the Amazon cloud, these steps are combined into one step—launching a specific instance type on Amazon EC2. The user is allocated a virtual machine that is ready to use for data and application setup.
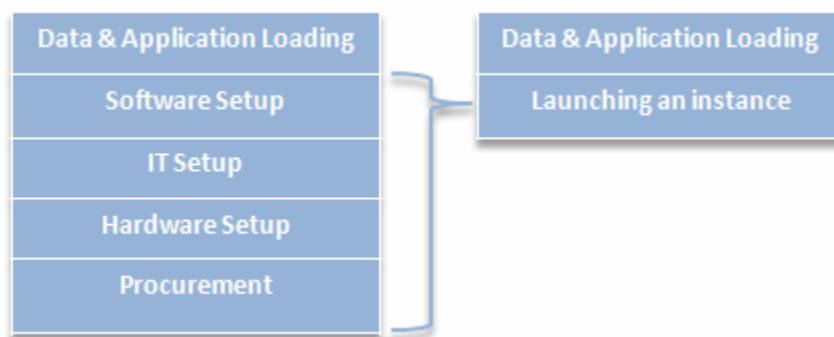
Time to Maintain

Setup time can be further reduced by creating an AMI, which contains applications, libraries, data, and associated configuration settings. The AMI can be used to launch a new instance. If at any time the instance fails, it can be terminated, and a new instance can be launched with no issues. This minimizes the time spent maintaining a machine after its initial launch.

Also, when new versions of software, updated applications, or data and maintenance patches are required, new AMIs can be created. The time used to create the new AMIs does not have to represent downtime for those who use the instances, as the AMI creation can be done simultaneously. Once the new AMI is available, it can be used immediately.

Choosing an Instance

Beyond the general concepts of data transfer, underutilized resources, and time, Amazon cloud consumers must consider the types of instances they are using as well as the other (optional) Amazon Web Services that they may add to their solution. Both instances and services are priced based on hourly use.

Amazon EC2 offers a number of predefined instance types from which to choose. Each instance type has a known configuration that defines the processing and memory capacity. Instance types are grouped into *families* such as Standard Instances, Micro Instances, High-Memory Instances, High-CPU Instances, and so on.

For example, within the Standard Instances family, a user can choose between small, large, and extra-large instances. The configuration of the large instance type has 7.5 GB of memory, four EC2 compute units (two virtual cores with two EC2 compute units each), 850 GB of local instance storage, and a 64-bit platform.

Note: The EC2 compute unit is generally equivalent to a 1.0–1.2 GHz 2007 Opteron or 2007 Xeon processor.

It's also worth noting that the size of the hard drive can be easily altered, as EBS drives can be attached and removed as required. Users will only pay for the gigabytes (GBs) they use. To see a complete list of each of the instance types and their configurations, visit aws.amazon.com/ec2/#instance.

Determining an instance type that provides the right capacity for applications that run in the cloud is similar to choosing the capacity of virtualized machines on premises.

On Amazon EC2, each instance type has an hourly cost that is dependent on the operating system and the software license associated with that instance. The ArcGIS Server AMI is preconfigured to run only with 64-bit instance types, and it is based on Windows 2008.

It is expected that Amazon cloud consumers will provide their own ArcGIS Server licenses to enable use of the Esri® preconfigured AMIs. As such, the hourly cost for running an instance on Amazon's infrastructure is based solely on the instance type and the license fee for using Windows.

For a complete list of the hourly cost per each instance type, visit aws.amazon.com/ec2/#pricing.

An Example Implementation

In this example, ArcGIS Server on Amazon EC2 is used to support a Web mapping application used by government officials, the public, and the media. Ideally, the application needs to be accessible 24 hours a day, seven days a week, and handle heavy traffic loads intermittently without degradation in performance. The IT and GIS staff want to be able to monitor cloud resource use and update the application and geodatabase on a quarterly basis. It is assumed that the organization has licensed ArcGIS Server.

The ArcGIS Server AMI is used to launch an instance that will publish and maintain three separate map services. The data used by the services is stored in a file geodatabase of about 80 GB in size, and the Web application is built on ArcGIS API for Flex™. Both the data and the application are also stored in the cloud as part of the instance.

Example Deployment Components

Components include

■ ArcGIS Server AMI
■ ArcGIS Server instance running

  ● File geodatabase
  ● Three published services
  ● Web application

■ Custom ArcGIS Server AMI
■ Amazon Elastic Load Balancer
■ Amazon CloudWatch plus Auto Scaling
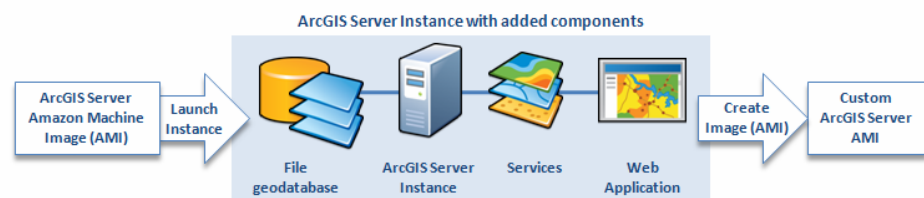
ArcGIS Server AMI

The ArcGIS Server instance is configured to include the application, the services, and the file geodatabase. The instance of ArcGIS Server is started using the Standard Large Instance type. This is the configuration from which additional production instances will be started.

Custom AMI
Once the ArcGIS Server instance is configured, it should be backed up. A custom AMI that is the exact copy of the newly reconfigured ArcGIS Server instance is created to preserve the configuration settings. This custom AMI can be used to restore/launch new instances if the production instances fail.

Note: Creating a custom AMI copies any EBS volumes that may have been attached. Be aware that this will affect costs when the custom AMI is deployed.

Elastic Load Balancer
In this type of scenario, it is usually necessary for the solution to automatically launch new instances of ArcGIS Server as they are needed, reduce instances when they are not needed, and have an Elastic Load Balancer (ELB) automatically unite the multiple instances under a common Web address. In this way, the solution can harness the power of all the instances under the ELB to respond to requests. The production instance is added to the ELB.

Note: Adding the ELB to a configuration will increase the costs when the solution is deployed.

Amazon CloudWatch plus Auto Scaling
Amazon CloudWatch can also be configured for the instances; it produces statistics and graphs describing the utilization of the instance resources. To add and remove instances automatically, the Amazon Auto Scaling API can be employed. The Auto Scaling service can be set to start and terminate instances according to specified criteria (e.g., when the processor average utilization exceeds 70% capacity for 10 minutes).

Note: Amazon CloudWatch will increase the costs of the solution. Auto Scaling is enabled by Amazon CloudWatch and carries no additional fees.
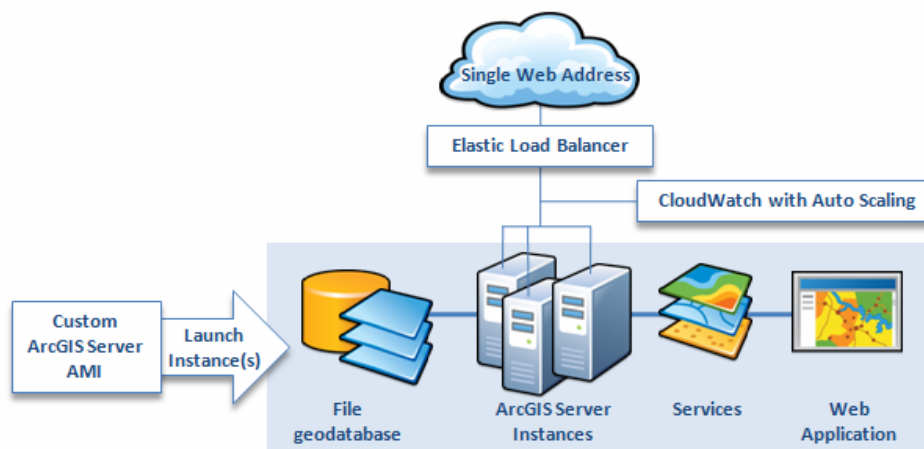
Update File Geodatabase
As data changes, the file geodatabase in the ArcGIS Server custom AMI may need to be updated to reflect those changes in the services and application. A new custom AMI that contains all the components, including the updated geodatabase, can be created. The new instances launched from the new AMI would have to be added to the ELB to replace the old instances.

Alternatively, EBS can be used to do data updates by directly updating the files on the EBS drive of a running instance. It's worth noting that in this case, every time a new instance is launched, the files on the EBS drives would need to be updated. Since the AMI will not include a backup of this update, snapshots can be used to keep a backup copy of only the EBS drive.

The choice for which option to use will depend on how often the data update is needed and the quantity of the data. If the data is updated often, the user might opt for the latter approach to minimize the amount of work and number of AMI versions.

Estimating Costs

Instance Costs

Using the example implementation, a cost per instance can be estimated. The hourly rate of the Standard Large Instance type is $0.48 per hour. Assuming 720 hours per month, the estimated cost per month for one instance would be $345.60. If the traffic to the site necessitated that the Auto Scaling feature maintain two instances at all times, then the cost per month would effectively double (i.e., $691.20).

US – N. Virginia	US – N. California	EU – Ireland	APAC – Singapore	
Standard On-Demand Instances		**Linux/UNIX Usage**	**Windows Usage**	
Small (Default)		$0.085 per hour	$0.12 per hour	
Large		$0.34 per hour	$0.48 per hour	
Extra Large		$0.68 per hour	$0.96 per hour	
High-Memory On-Demand Instances				
Extra Large		$0.50 per hour	$0.62 per hour	
Double Extra Large		$1.20 per hour	$1.44 per hour	
Quadruple Extra Large		$2.40 per hour	$2.88 per hour	
High-CPU On-Demand Instances				
Medium		$0.17 per hour	$0.29 per hour	
Extra Large		$0.68 per hour	$1.16 per hour	
Cluster Compute Instances				
Quadruple Extra Large		$1.60 per hour	N/A*	

* Windows is not currently available for Cluster Compute Instances.

Please check for updated prices at aws.amazon.com/ec2/.

In some cases, additional instances are needed only during peak times. For example, a spike in traffic may occur due to an overwhelmingly popular public event, with the increase in traffic expected to last throughout that day, requiring an additional instance during that time. These situations can be scheduled and may help reduce costs.

If access to the application and/or services does not require high availability, it may not be necessary to plan for multiple instances. Additionally, an instance can be shut down during certain low-activity hours (e.g., at night or on the weekends and holidays).

Keep in mind that during normal operations, additional instances may need to be started for reasons other than Web traffic or increased load on the server. For example, troubleshooting, testing, updates, or creating a new AMI may require adding more

instances, at least for a temporary period. Planning some margin cost for such activities would be considered a best practice.

Amazon EC2 Reserved Instances are another way to save costs. If it is known that an instance will be used all the time for a full year, Reserved Instances can be requested and the cost prepaid at a discounted rate. For example, given a Standard Large Reserved Instance running full-time for one year (8,760 hours) at $0.20 per hour, the cost would be $1,752, plus the $910 one-time fee, for a total of $2,662. This is a savings of $1,542 from the On-Demand Instance equivalent scenario.

US – N. Virginia	US – N. California		EU – Ireland	APAC – Singapore
One-time Fee				
Standard Reserved Instances	1 yr Term	3 yr Term	Linux/UNIX Usage	Windows Usage
Small (Default)	$227.50	$350	$0.03 per hour	$0.05 per hour
Large	$910	$1400	$0.12 per hour	$0.20 per hour
Extra Large	$1820	$2800	$0.24 per hour	$0.40 per hour
High-Memory Reserved Instances				
Extra Large	$1325	$2000	$0.17 per hour	$0.24 per hour
Double Extra Large	$3185	$4900	$0.42 per hour	$0.55 per hour
Quadruple Extra Large	$6370	$9800	$0.84 per hour	$1.10 per hour
High-CPU Reserved Instances				
Medium	$455	$700	$0.06 per hour	$0.125 per hour
Extra Large	$1820	$2800	$0.24 per hour	$0.50 per hour
Cluster Compute Reserved Instances				
Quadruple Extra Large	$4290	$6590	$0.56 per hour	N/A*
* Windows is not currently available for Cluster Compute Instances.				

Please check for updated prices at aws.amazon.com/ec2/.

Storage and Data Transfer Costs   Input and output costs are measured by size in gigabytes. Input and output costs will go down with higher usage. This area is among the hardest to estimate, since it will depend on each individual service and the way it's configured with the content. It will also depend on how many users make requests to the instances at a certain time. It is recommended that, if the data size is going to be large (e.g., a terabyte or more), users do some testing and benchmarks to understand the expected costs for data transfer.

Internet Data Transfer

The pricing below is based on data transferred "in" and "out" of Amazon EC2.

Data Transfer In	US & EU Regions	APAC Region
All Data Transfer	Free until Nov 1, 2010 *	Free until Nov 1, 2010 *

Data Transfer Out **	US & EU Regions	APAC Region
First 1 GB per Month	$0.00 per GB	$0.00 per GB
Up to 10 TB per Month	$0.15 per GB	$0.19 per GB
Next 40 TB per Month	$0.11 per GB	$0.15 per GB
Next 100 TB per Month	$0.09 per GB	$0.13 per GB
Over 150 TB per Month	$0.08 per GB	$0.12 per GB

* Data Transfer In will be $0.10 per GB after November 1, 2010.
** Rate tiers take into account your aggregate Data Transfer Out usage across Amazon EC2, Amazon S3, Amazon RDS, Amazon SimpleDB, Amazon SQS, Amazon SNS, and Amazon VPC.

There is no Data Transfer charge between Amazon EC2 and other Amazon Web Services within the same region (i.e. between Amazon EC2 US West and Amazon S3 in US West). Data transferred between Amazon EC2 instances located in different Availability Zones in the same Region will be charged Regional Data Transfer. Data transferred between AWS services in different regions will be charged as Internet Data Transfer on both sides of the transfer.

Usage for other Amazon Web Services is billed separately from Amazon EC2.

Availability Zone Data Transfer

$0.00 per GB – all data transferred between instances in the same Availability Zone using private IP addresses.

Regional Data Transfer

$0.01 per GB in/out – all data transferred between instances in different Availability Zones in the same region.

Prices vary and change regularly. Please check updated prices at aws.amazon.com/ec2/.

Elastic Block Store Costs

EBS storage costs are calculated using the amount of input/output and the size of the storage in gigabytes. However, in this case, the input/output is calculated based on the number of requests, not their size. Since this is considered disk reads, it's usually good to be quite generous with the number of requests estimated.

Using the monthly bill to track the costs of input/output is another good way to understand the impact of the costs. Some limited usage can serve as a benchmark of what the bill might be, and from there, certain estimates can be made on how it may grow with increased usage.

Most users try to limit the size of the EBS to only what is needed for supporting the solution. An EBS volume can be attached to only one instance. When a new instance is started, a new EBS volume is started as well.

US – N. Virginia	US – N. California	EU – Ireland	APAC – Singapore

Amazon EBS Volumes

$0.10 per GB-month of provisioned storage
$0.10 per 1 million I/O requests

Amazon EBS Snapshots to Amazon S3 (priced the same as Amazon S3)

$0.15 per GB-month of data stored
$0.01 per 1,000 PUT requests (when saving a snapshot)
$0.01 per 10,000 GET requests (when loading a snapshot)

Prices vary and change regularly. Please check updated prices at aws.amazon.com/ec2/.

Snapshots are used as backups of the EBS drive. There's a different rate for storing snapshots. A snapshot can be used to create the second EBS volume rather than having to create both volumes from scratch.

Elastic IP Address Costs

Using an elastic IP address service does not include any extra costs unless it is not attached to an instance. The best practice is to keep an elastic IP address attached to an instance and in use.

No cost for Elastic IP addresses while in use

$0.01 per non-attached Elastic IP address per complete hour

$0.00 per Elastic IP address remap – first 100 remaps / month

$0.10 per Elastic IP address remap – additional remap / month over 100

Prices vary and change regularly. Please check updated prices at aws.amazon.com/ec2/.

Elastic Load Balancing Costs

Elastic Load Balancing costs are charged based on two components—the number of hours the ELB is running and the amount of data (per GB) that was processed through the ELB as input and output. In this case, it's safe to assume that the input and output of the ELB is equal to all the collective external input and output of the instances registered with it.

US – N. Virginia	US – N. California	EU – Ireland	APAC – Singapore

$0.025 per Elastic Load Balancer-hour (or partial hour)
$0.008 per GB of data processed by an Elastic Load Balancer

Prices vary and change regularly. Please check updated prices at aws.amazon.com/ec2/.

CloudWatch Costs

CloudWatch costs $0.015 per instance per hour regardless of the instance size when using detailed monitoring at a one-minute frequency. So if an instance is running for 720 hours, as in the earlier assumption, the cost for monitoring it every month would be $10.80 per instance. Basic monitoring at a five-minute frequency is free.

Prices vary and change regularly. Please check updated prices at aws.amazon.com/ec2/.

Amazon Calculators

Amazon provides a calculator to help users determine their potential monthly costs when using some of the services; it can be found at calculator.s3.amazonaws.com/calc5.html.

Another calculator is also available as a spreadsheet and can help compare costs between different options for infrastructure. It's available at awsmedia.s3.amazonaws.com/Amazon_EC2_Cost_Comparison_Calculator_022210.xls.

The spreadsheet contains many options to compare between using Amazon, a colocation facility, and traditional on-premises data center costs. Users can fine-tune the settings within the spreadsheet for each of those options depending on their work to model their annual total cost of ownership (TCO).

Amazon EC2 Bill Sample

The bill for using Amazon EC2 shows the costs of the instances based on use plus the cost of the EBS drives attached to those instances, as well as other AWS services used (e.g., Elastic Load Balancing, CloudWatch).

Amazon Elastic Compute Cloud		
View/Edit Service		
US East (Northern Virginia) Region		
Amazon EC2 running Windows		
$0.48 per Large Windows Instance (m1.large) instance-hour (or partial hour)	2,098 Hrs	1,007.04
Amazon EC2 EBS		
$0.10 per GB-month of provisioned storage	1,047.867 GB-Mo	104.79
$0.10 per 1 million I/O requests	11,394,333 IOs	1.14
$0.15 per GB-Month of snapshot data stored	8.767 GB-Mo	1.32
$0.01 per 10,000 gets (when loading a snapshot)	103,623 Requests	0.10
$0.01 per 1,000 puts (when saving a snapshot)	2,488 Requests	0.02
Elastic Load Balancing		
$0.008 per GB Data Processed by the LoadBalancer	4.048 GB	0.03
$0.025 per LoadBalancer-hour (or partial hour)	1,057 Hrs	26.43
Amazon CloudWatch		
$0.015 per monitored instance-hour (or partial hour)	1,171 Hrs	17.57
	Download Usage Report »	**1,158.43**

Overall costs are represented as a combination of Amazon service reports.

Billing Statement: June 1, 2010

The billing cycle for this report is May 1 - May 31, 2010.

Expand All Services | Collapse All Services		Printer Friendly Version
		Totals
Amazon Elastic Compute Cloud		
View/Edit Service		
	Download Usage Report »	1,158.43
Amazon Simple Storage Service		
View/Edit Service		
	Download Usage Report »	0.00
Amazon Virtual Private Cloud		
View/Edit Service		
	Download Usage Report »	0.00
AWS Data Transfer (excluding Amazon CloudFront)		
View/Edit Service		
		0.63
Taxes		0.00
Total Charges due on June 1, 2010		**$1,159.06**

Summary   Planning for the types of loads, usage, and features, along with the costs of each of the cloud services, provides a way to estimate the potential costs of cloud services.

For ArcGIS Server on Amazon EC2

- Begin with a Standard Large Instance type as a baseline for estimating instance costs.
- For solutions requiring high availability, plan for multiple instances and costs associated with Elastic Load Balancing and Amazon CloudWatch.
- While an elastic IP address will not necessarily add to the cost of the solution (as long as it is attached to an instance), it makes instance replacement seamless and easy if an instance goes down or needs to be replaced. Without the elastic IP address, troubleshooting and setup costs could impact the project.
- Creating a custom AMI that reflects the solution setup can save time and money when it becomes necessary to re-create or update the setup.

It is good practice to compare estimates to the actual bill to see how accurate those estimates are. Continuous monitoring of the system and the bills is one of the best tools to keep costs under control.

For details on how to set up and use the ArcGIS Server AMIs, please refer to the ArcGIS Server on Amazon EC2 documentation at help.arcgis.com/en/arcgisserver/10.0/help/arcgis_server_on_amazon_ec2/.

About Esri

Since 1969, Esri has been helping organizations map and model our world. Esri's GIS software tools and methodologies enable these organizations to effectively analyze and manage their geographic information and make better decisions. They are supported by our experienced and knowledgeable staff and extensive network of business partners and international distributors.

A full-service GIS company, Esri supports the implementation of GIS technology on desktops, servers, online services, and mobile devices. These GIS solutions are flexible, customizable, and easy to use.

Our Focus

Esri software is used by hundreds of thousands of organizations that apply GIS to solve problems and make our world a better place to live. We pay close attention to our users to ensure they have the best tools possible to accomplish their missions. A comprehensive suite of training options offered worldwide helps our users fully leverage their GIS applications.

Esri is a socially conscious business, actively supporting organizations involved in education, conservation, sustainable development, and humanitarian affairs.

Contact Esri

1-800-GIS-XPRT (1-800-447-9778)
Phone: 909-793-2853
Fax: 909-793-5953
info@esri.com
esri.com

Offices worldwide
esri.com/locations

380 New York Street
Redlands, CA 92373-8100 USA