

Concepts for ArcGIS Server 10.1

What is ArcGIS for Server?

[Services](#) » [ArcGIS for Server \(Windows\)](#)

ArcGIS for Server is software that makes your geographic information available to others in your organization and optionally anyone with an Internet connection. This is accomplished through web services, which allow a powerful server computer to receive and process requests for information sent by other devices. ArcGIS for Server opens your GIS to tablets, smartphones, laptops, desktop workstations, and any other devices that can connect to web services.

To get started with ArcGIS for Server, you'll need to prepare your hardware, software, and data, then you can set up GIS web services. Finally, you can use various types of applications to consume your services.

Preparing hardware, software, and data

The hardware you use for your server is typically more powerful than your other desktop computers. ArcGIS 10.1 for Server requires a machine capable of running a 64-bit operating system. The ArcGIS for Server architecture is scalable, meaning you can add multiple machines if extra processing power is needed.

Depending on organizational requirements, you may need the help of your IT staff to allow your server to be accessed over the Internet. When planning your hardware and environment, remember that ArcGIS for Server can also be deployed on virtual machines or commercial cloud platforms such as Amazon EC2.

Once you install ArcGIS for Server, you can start using it right away or you can integrate it with your organization's existing web server by installing the ArcGIS Web Adaptor. You also need to have ArcGIS for Desktop on at least one computer in your organization in order to publish GIS web services. This computer does not have to be the server.

Publishing GIS web services

If you've worked with ArcGIS for Desktop, then you know how to use applications like ArcMap and ArcGlobe to view and analyze GIS data. You use these same applications when publishing web services to ArcGIS for Server. You can author maps, geoprocessing models, mosaic datasets, and other GIS resources in ArcGIS for Desktop and use a simple wizard to share them as web services.

As part of the sharing process, ArcGIS alerts you to potential performance issues in the resource you are publishing. It also checks its list of registered data locations to understand whether it needs to fix any paths after your resource is moved to the server.

Below are the types of resources you can publish to ArcGIS for Server:

GIS resource	What it can do in ArcGIS for Server	Which ArcGIS for Desktop application creates it
Map document	Mapping, network analysis, Web Coverage Service (WCS) publishing, Web Feature Service (WFS) publishing, Web Map Service (WMS) publishing, Web Map Tile Service (WMTS) publishing, mobile data publishing, KML publishing, Geodatabase data extraction and replication, feature access publishing, schematics publishing	ArcMap
Address locator	Geocoding	ArcCatalog or the Catalog window in ArcMap
Geodatabase	Geodatabase query, extraction, and replication; WCS publishing; WFS publishing	ArcCatalog or the Catalog window in ArcMap
Geoprocessing model or tool	Geoprocessing, Web Processing Service (WPS) publishing	ArcMap (geoprocessing result from the Results window)
ArcGlobe document	3D mapping	ArcGlobe
Raster dataset or mosaic dataset or layer file referencing a raster dataset or mosaic dataset	Image publishing, WCS or WMS publishing	ArcCatalog or the Catalog window in ArcMap
Folders and geodatabases of GIS content	Create a searchable index of your organization's GIS content	ArcMap

If you don't want to publish right away (for example, if you don't have immediate access to the server machine) you can save a service definition file instead and publish it later. The service definition includes all the data paths and properties necessary to publish the service at another time. You can even choose to include all the source data, allowing you to truly package the service into one transferrable file.

During the publishing process, you'll enable capabilities that define the various ways your audience can use the service. For example, Feature Access is a popular capability that allows web users to edit vector features in a map service. Another example of a capability is WMS, which exposes your service through the Open Geospatial Consortium (OGC) Web Map Service (WMS) specifications.

See [What types of services can you publish](#) for greater detail on the services and capabilities available to you.

If you find that your web service does not offer the precise functionality or business logic that you need, you can expand it through server object extensions (SOEs). An SOE extends the base functionality of a web service using ArcObjects, the vast suite of components on which the Esri family of products is built. SOEs are an advanced option requiring custom development, but once written they are easy to deploy to your server or share it with others. No special software other than ArcGIS for Server is required to run an SOE.

Using GIS web services

Once you have web services running, you can use them in any application, device, or API that can communicate through HTTP (Hyper Text Transfer Protocol).

- The ArcGIS.com map viewer lets you make and save online maps that display your services. You can optionally overlay your service with other services and save your map in ArcGIS Online, Esri's cloud-hosted repository of online content.
- The [ArcGIS Viewer for Flex](#) and the [ArcGIS Viewer for Silverlight](#) offer interactive wizards for building aesthetically pleasing, functional web mapping applications with your services.
- The ArcGIS APIs for JavaScript, Flex, Silverlight, iOS, Android, and Windows Phone allow you to develop custom applications that use the full breadth of your web services within an interface you design yourself.
- ArcGIS for Desktop applications, such as ArcMap and ArcGlobe, are designed to use web services published by ArcGIS for Server. Using a service in these applications is often as easy as clicking the Add Data button.
- Any other application that can make a SOAP or REST web service request can connect to ArcGIS for Server. Supported clients range from smartphone and tablet apps that find the nearest grocery store to enterprise desktop applications for customer management or resource planning.

Maintaining your server

As you work with your server over time, you'll need to adjust settings, add and remove services, and set up security rules. ArcGIS Server Manager is a web application included with every installation of ArcGIS for Server that provides an intuitive point-and-click interface for administering the server. You can use

Manager to view the server logs, stop and start services, publish service definitions, define users and roles for security, and perform other similar tasks.

As easy as it is to use Manager, there are times when you may want to administer your server automatically through scripting. ArcGIS for Server has a REST-ful administrator API that allows you to automate server management tasks using the scripting language of your choice. For example, you can write a Python script that checks the health of your services periodically and sends you an e-mail if a service is detected to be down. This help system contains various [examples](#) of how to script your server administration.

Summary

ArcGIS for Server opens your GIS to many types of devices through web services. You can take the resources you're familiar with through ArcGIS, such as map documents and geoprocessing models, and publish them to your server to create GIS web services. The services can be consumed in any application or device that can make a web service call using HTTP.

ArcGIS for Server includes an administrative Manager application and an API which you can use to maintain your server and set rules for security, logging, and other behavior.

The help system you are reading contains more detail on all of the subjects above.

Site Planning: Servers: Deployment scenarios

[Services](#) » [ArcGIS for Server \(Windows\)](#) » [Administering ArcGIS for Server](#) » [Planning and maintaining an ArcGIS Server site](#)

This topic describes different ways you can architect your ArcGIS Server site to meet various capacity and availability requirements.

The following terms are used to help explain each deployment scenario:

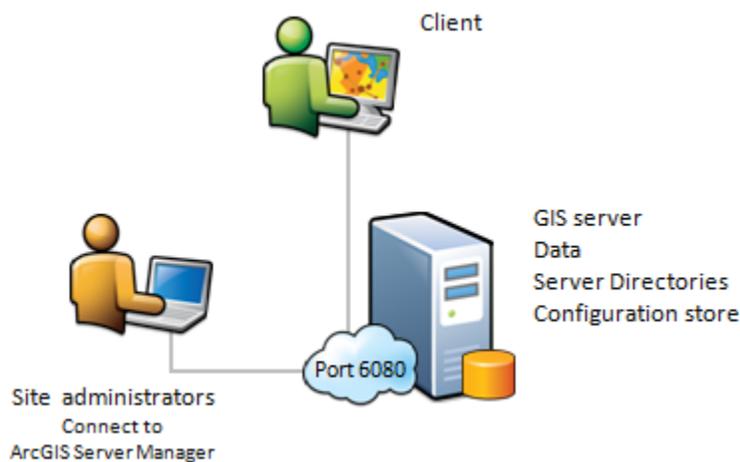
- **Site:** A site consists of several components, such as a GIS server and ArcGIS Web Adaptor, that can optionally be distributed across multiple machines to increase computing power and redundancy. For a more detailed description, see [Inside an ArcGIS Server site](#).
- **GIS server:** The main component of the site that does the work to satisfy requests issued to the GIS web services. A GIS server can draw maps, run tools, serve imagery, and perform many other operations offered by ArcGIS.
- **ArcGIS Web Adaptor:** An optional component that allows you to configure a web entry point into your site. It integrates with your web server and distributes incoming requests among GIS servers. For more information, see [About the ArcGIS Web Adaptor](#).

- Server directories: A set of directories containing certain types of files that support your services. These files include caches, search indexes, and geoprocessing job results. For more information, see [About server directories](#).
- Configuration store: A location that contains configuration information such as the list of GIS servers participating in the site. The configuration store must be available for your site to function. For more information, see [About the configuration store](#).
- Data: Data supporting your web services, such as feature classes, tools, imagery, and locators. For more information, see [Making your data accessible to ArcGIS Server](#).

The following scenarios are presented as guides for you to consider as you build your ArcGIS Server site. Although you could configure your site exactly as presented in one of the scenarios, these configurations are flexible and can be adjusted to fit specialized needs and hardware resources.

Developer sandbox site

When you're just developing or experimenting with ArcGIS Server, you can simply install the GIS server without installing a web server or the Web Adaptor.



Developer sandbox site containing one GIS server. Data, server directories, and the configuration store reside locally on the GIS server.

In this scenario, the site is configured with one GIS server. The data, server directories, and configuration store reside locally on the GIS server. A Microsoft SQL Server Express database is a good option for setting up a small instance of a geodatabase on the GIS server.

Clients access the developer sandbox site by connecting directly to the GIS server through HTTP on port 6080. For example, the URL to the site would be `http://myserver:6080`. The GIS server hosts services only; there is no web server in this configuration for hosting web applications.

Use cases and advantages of the developer sandbox site

This configuration is ideal for prototyping services and isolated sandbox testing. It is relatively simple to install and maintain.

Disadvantages of the developer sandbox site

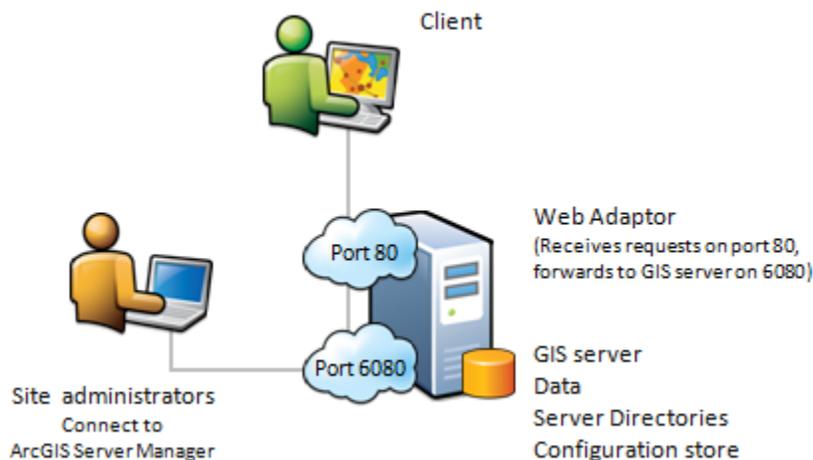
This configuration is not very secure, since ArcGIS Server Manager and the ArcGIS Server Administrator Directory are exposed through the same port that everyone else uses to access the services. Also, this configuration cannot host web applications, and there is no failover option if the GIS server goes offline.

Single machine site

The simplest appropriate configuration for a production site is to expose one GIS server through the Web Adaptor.

The Web Adaptor is recommended so that incoming requests can go through your established web server. This gives you more security options and the ability to host web applications. If you are short on resources or you don't have to accommodate many concurrent requests, you can install both the GIS server and Web Adaptor on a single machine. This machine must also have a web server installed.

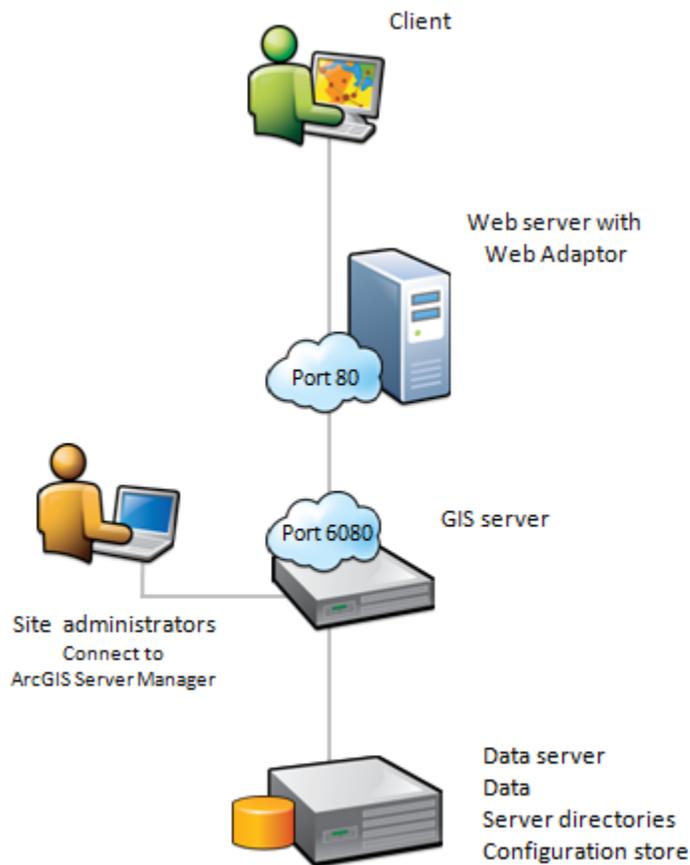
For example, the site in the diagram below is configured with a Web Adaptor on port 80 and would be accessed using the URL <http://myserver>. The Web Adaptor forwards incoming client requests to the GIS server on port 6080. Server administrators should log in to Manager or the Administrator Directory through port 6080.



Single machine site with the Web Adaptor installed on the GIS server.

You can architect your site to use parts of your organization's existing IT infrastructure. In the diagram below, the Web Adaptor has been offloaded to a web server on a separate machine. Similarly, the data,

server directories, and configuration store have been put on a dedicated data server. This shows that the phrase "single machine site" technically means "single GIS server site."



Site with one GIS server with the Web Adaptor and data offloaded to separate machines.

Putting the web server on its own machine may be desirable in organizations where the web server has a different set of administrators or access policies than the GIS server.

Putting the data on a separate machine allows you to add and remove GIS servers from your site without any disruption to your data path settings. Putting the server directories and configuration store on a redundant network storage device improves your abilities to back up and recover these resources.

Use cases and advantages of the single machine site

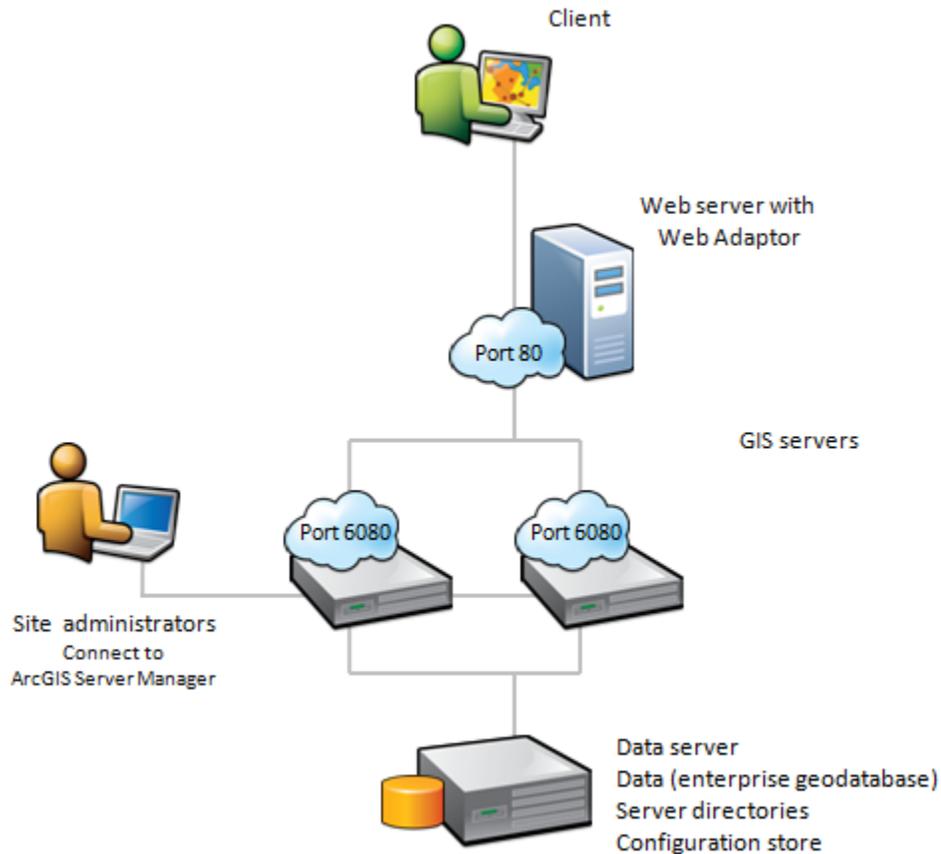
The single machine site as diagrammed above with a Web Adaptor is ideal for accommodating small numbers of concurrent users. It's also useful in development or prototyping scenarios where extra security or the ability to host web applications is desired. The single machine site is relatively simple to configure and can integrate into your existing web server and data storage architecture.

Disadvantages of the single machine site

The single machine site has no failover abilities if the GIS server goes offline. Also, the capacity of the GIS server is limited to the physical hardware characteristics of the single machine.

Multiple machine site

A site can include multiple GIS servers to handle increased traffic or provide a backup in case one of the GIS servers goes offline. The following diagram shows the simplest way to configure a site with multiple GIS servers. The Web Adaptor detects the GIS servers that participate in the site and forwards requests to each in a round-robin fashion. The GIS servers also perform some degree of request distribution among themselves.

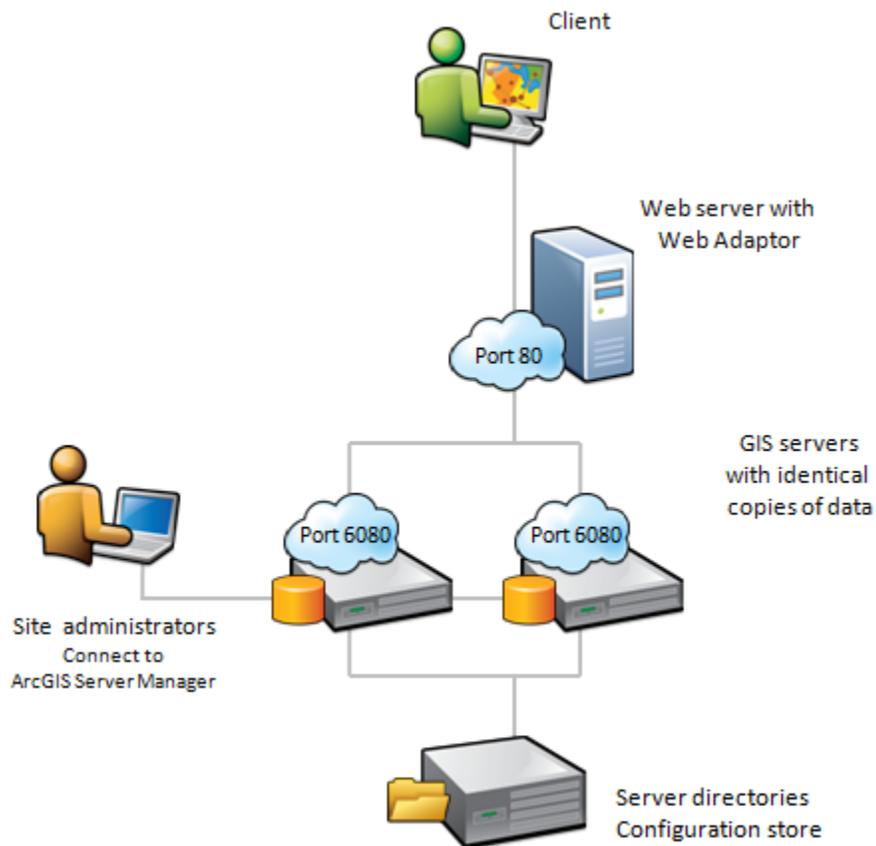


Site with multiple GIS servers where the data resides on a highly available data server.

There are two strategies for storing the data when using multiple GIS servers. The approach shown above keeps the data in one centralized location visible to each GIS server. The data only has to be maintained in one place, and this configuration is recommended if you have a good intranet connection.

The other approach for data storage, shown below, is to put a local copy of the data on each GIS server machine at an identical path. This strategy cuts down on network calls and can increase performance if

your intranet connection speed is slow. However, it's difficult to maintain large, frequently changing datasets with this architecture.

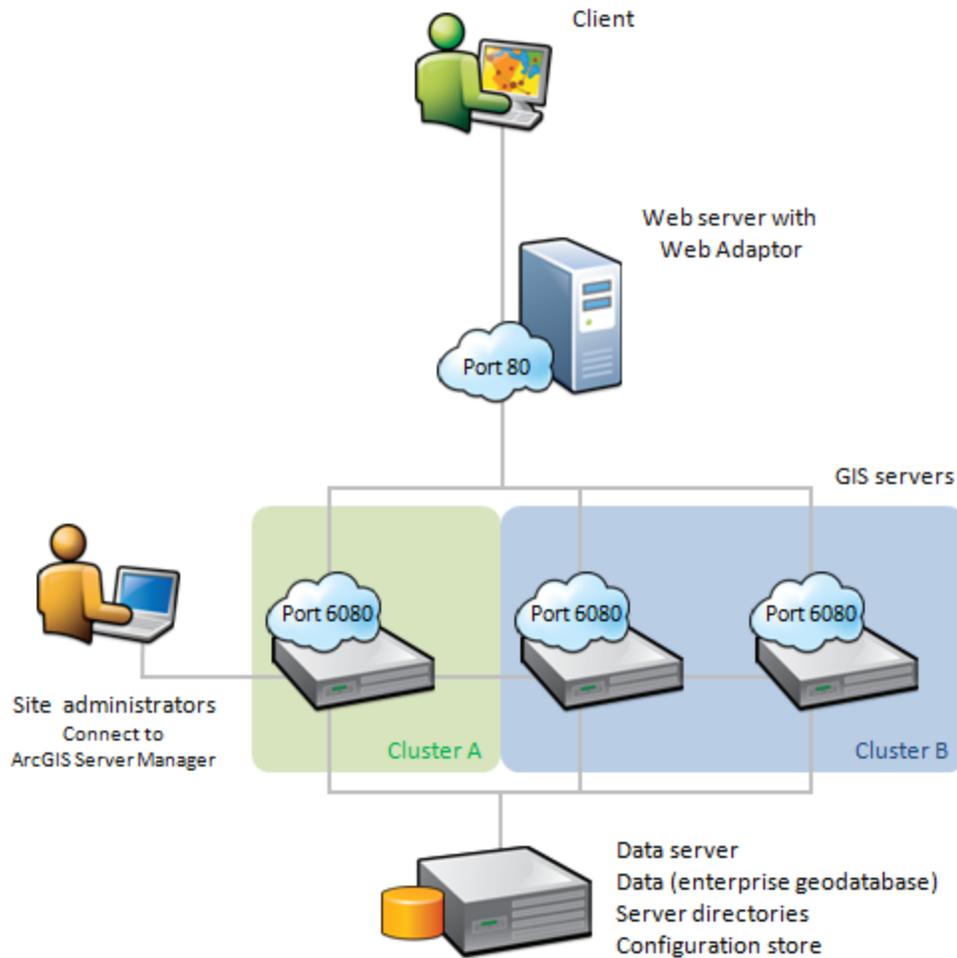


Site with multiple GIS servers where data is stored locally on each GIS server.

If demand increases in either of the above scenarios, additional GIS server machines can be added to the site either manually or automatically (through scripts). This architecture is well suited to cloud computing, in which any GIS server can be added or removed from the site at any time.

Taking advantage of clusters

Large sites with two or more GIS servers can take advantage of clusters. A cluster is a group of GIS servers that has been configured to run a dedicated subset of services. In the diagram below, Cluster A could potentially be configured to run map services, while Cluster B (with higher processing power) could be configured to run geoprocessing services.



Multiple machine site with clusters. Each cluster runs its own subset of services.

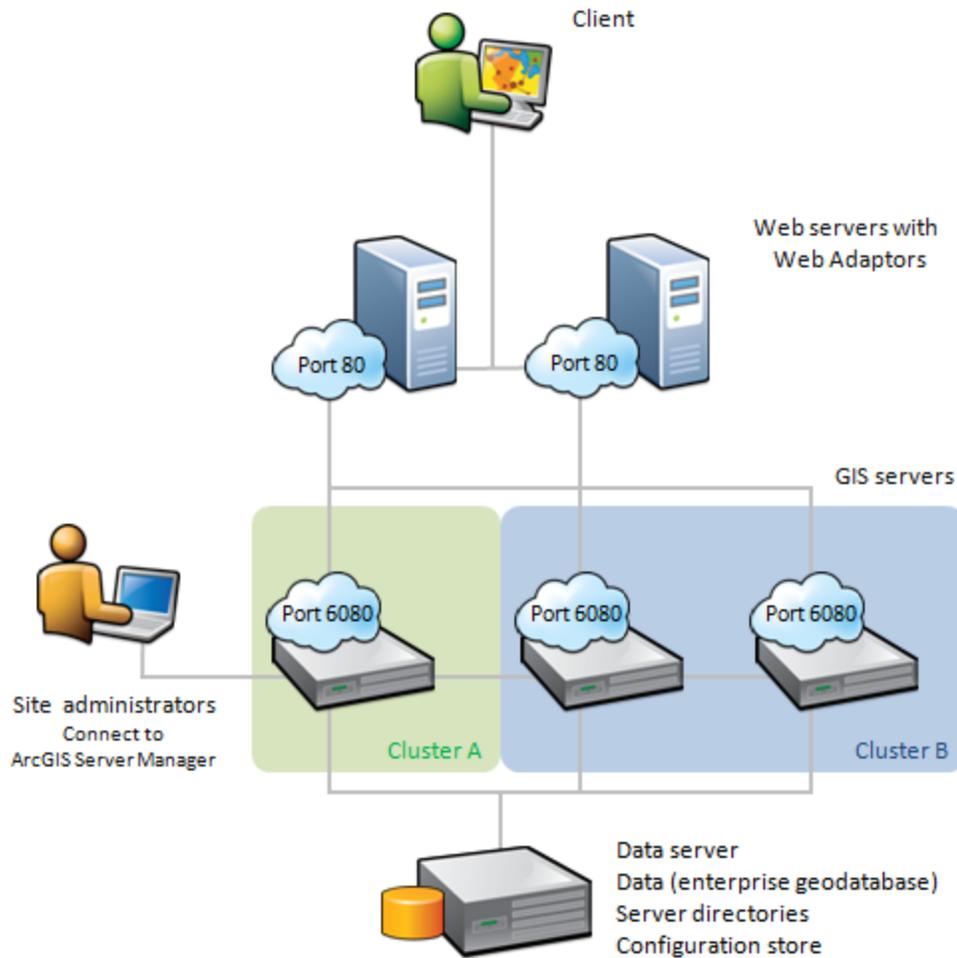
Some server operations, such as batch geocoding, are very CPU intensive. Using clustered servers for this type of operation may help free up other machines in your site so that your remaining services can remain online unencumbered.

Clustering is also useful when you have disparate hardware resources. For example, an older or slower server could be placed in its own cluster to perform lower-priority jobs.

For more information, see [About GIS server clusters](#).

Using more than one web server

To help ensure high availability of your site, you can also establish redundancy at the web server tier. In the diagram below, two web servers installed with the Web Adaptor act as identical entry points into your site on port 80. This helps keep your site running in the event of an unplanned outage on one of the web servers. It can also help reduce the load on the first web server machine.



Site with redundancy at the web server tier. The clusters are optional.

Following are several other scenarios that might require two Web Adaptors, although these are less common:

- You need internal users to connect to Manager through the Web Adaptor, but you want to block external users from accessing the Manager login screen. This could occur if ArcGIS Server is in a perimeter network (DMZ) and your firewalls block it from receiving connections from your internal network through port 6080. In this case you set up a second Web Adaptor for internal users only that communicates through whichever port is allowed by your firewall.
- You want to block external users from accessing the Manager login screen, but you are also performing web tier authentication. In this case, you set up a second Web Adaptor for internal users only that allows publishing and administrative access.

Use cases and advantages of the multiple machine site

The multiple machine site is ideal for enterprise deployments that need to accommodate more users than a single machine can handle. This architecture can be scaled out to include as many machines as needed, thereby multiplying the processing power of the site. GIS servers can even be added in response to user demand. This is useful in cloud environments like Amazon EC2 that offer automatic scaling based on usage statistics.

The multiple machine site is also appropriate for sites that cannot afford to have any downtime. If a GIS server goes offline, the other GIS servers can keep the site running.

Disadvantages of the multiple machine site

The multiple machine site requires an extra degree of setup and obviously more hardware resources. Because the site is able to gracefully continue running if a GIS server goes offline, the server administrator must set up his or her own monitoring or alert schedule to understand whether a machine has become unavailable.

Summary

ArcGIS Server is designed to accommodate both large and small deployments. When you first begin building your site, you might want to start small and install all the components on a single machine. As you get ready to deploy your production site, or if you need to handle more users, you can add more GIS servers. You can also integrate your site into your existing IT infrastructure by using your own enterprise web server (via the Web Adaptor) or data server. Finally, many of the components in the ArcGIS Server architecture can be duplicated or run in parallel to avoid a single point of failure.

System requirements

[Step 2: Verifying system requirements](#)

The system specifications, including hardware requirements, supported operating systems, and web servers, that are required to run ArcGIS for Server are available at [ArcGIS for Server system requirements](#).

Supported operating systems

 Note: 32-bit operating systems are no longer supported. The ArcGIS for Server setup will only proceed if the operating system is 64-bit.

The operating systems below satisfy the minimum operating system requirements:

- Windows Server 2003 Service Pack 2 Standard, Enterprise, and Datacenter 64bit (EM64T)
- Windows Server 2008 Standard, Enterprise, and Datacenter 64-bit (EM64T)
- Windows Server 2008 R2 Standard, Enterprise, and Datacenter 64-bit (EM64T)

- Windows Server 2012 Standard and Datacenter 64-bit (EM64T)
- Windows 8 (Basic), Pro, and Enterprise (EM64T)
- Windows 7 Ultimate and Professional 64-bit (EM64T)
- Windows Vista Service Pack 2 Ultimate, Enterprise, Business 64-bit (EM64T)
- Windows XP Service Pack 2 Professional Edition 64-bit (EM64T)

Disk space requirements

ArcGIS for Server requires a minimum of 2,000 MB of available disk space, in which 350 MB of this total space must be available on the system drive.

Additional system requirements

- Machines with an underscore (_) in their names are not supported. Several widely-used Internet hostname specifications have designated the underscore character as nonstandard. Although Windows allows you to use the underscore in a machine name, it can still cause problems when you interact with other servers and platforms. For this reason, ArcGIS for Server will not proceed with installation on servers that have an underscore in the hostname.
- ArcGIS for Server is not supported on domain controllers. Installing ArcGIS for Server on a domain controller may adversely affect functionality.

Ports

ArcGIS for Server uses certain ports to communicate with machines on the Internet and intranet. For more information, see [Ports used by ArcGIS for Server](#).

Microsoft .NET Framework requirement for .NET Extension Support feature

The .NET Extension Support feature requires Microsoft .NET Framework 3.5 Service Pack 1 (SP1) . If Microsoft .NET Framework 3.5 SP1 is not found, the .NET Extension Support feature will not be available for installation. Microsoft .NET Framework 3.5 SP1 is available on the ArcGIS for Server media. It is also available for download from the Microsoft website. However, on Windows 7 and Windows Server 2008 R2, Microsoft .NET Framework 3.5 Service Pack 1 must be installed using Windows Features. Follow the instructions below to install Microsoft .NET Framework 3.5 SP1 on Windows 7 and Windows Server 2008 R2:

Windows 7

Go to Start > Control Panel > Programs > Turn Windows Features on or off. Enable and install Microsoft .NET Framework 3.5.1.

Windows Server 2008 R2

Open Server Manager > Features > Add Features. Enable and install Microsoft .NET Framework 3.5.1.

Microsoft Core XML Services (MSXML) 6

ArcGIS for Server requires Microsoft Core XML Services (MSXML) 6. The ArcGIS for Server installation will not proceed if MSXML 6 is not found on the machine. If the ArcGIS for Server installation is performed through the GUI, the installation process automatically installs it for you. If you are installing ArcGIS for Server silently or in certain deployment environments, MSXML 6 must be installed separately. The MSXML 6 setup is available at <ArcGIS for Server Installation folder>\Support\MSXML6\64-bit\msxml6_x64.msi.

ArcGIS for Server geoprocessing requirements

Python 2.7, Numerical Python 1.6.1, and Matplotlib 1.1.0 are required to support certain core geoprocessing tools. It is recommended that Python 2.7, Numerical Python 1.6.1, and Matplotlib 1.1.0 are installed by the ArcGIS for Server setup. If Python 2.7 and Numerical Python 1.6 are not detected, the ArcGIS for Server setup will automatically install them.

Web browser requirements

Microsoft Internet Explorer 7 or higher is required. You must obtain and install Internet Explorer 7 or higher prior to installing ArcGIS for Server.

ArcGIS Server Manager requires one of the following web browsers to be installed:

- Google Chrome (version 10 or higher)
- Mozilla Firefox (version 3.5 or higher)
- Microsoft Internet Explorer (version 7, 8, 9)

Caution:

It is recommended that you use ArcGIS Server Manager in either a Firefox, Chrome, or Internet Explorer 9 web browser to have access to the best possible user experience. Using Internet Explorer 7 and 8 may result in a lower-quality display of ArcGIS Server Manager.

Interaction between ArcGIS for Server and ArcGIS products before 10.1

If the following products are installed on your machine and are a version before 10.1, they must be uninstalled before proceeding with the ArcGIS for Server setup:

- ArcGIS Desktop
- ArcInfo Workstation
- ArcReader stand-alone

- ArcIMS
- ArcIMS Web ADF for the Java Platform
- ArcIMS Web ADF for the .NET Framework
- ArcGIS Server for the Java Platform
- ArcGIS Server for the .NET Framework
- ArcGIS Server Web ADF Runtime for the .NET Framework
- ArcGIS Engine Runtime
- ArcGIS Engine Developer Kits
- ArcGIS Desktop Developer Kits
- ArcGIS Image Server (Including all clients)

The ArcGIS for Server setup displays a dialog box if conflicting versions of these products exist on your machine. If so, you must uninstall these products then run the ArcGIS for Server setup again to proceed with your installation.

These products can exist together on the same machine if they are all version 10.1.

How ArcGIS Server security works

[Services](#) » [ArcGIS for Server \(Windows\)](#) » [Administering ArcGIS for Server](#)

ArcGIS Server comes with a robust and effective security framework. By configuring ArcGIS Server security, you can manage and control access to ArcGIS Server. ArcGIS Server allows you to control who can administer ArcGIS Server, who can publish GIS services to it, and who can access hosted GIS web services.

Users, roles, and permissions

When a resource on ArcGIS Server is secured, only authorized users are allowed to access that resource. ArcGIS Server manages access to a secured resource by using a role-based access control system. There are three main components in a role-based access control system: users, roles, and permissions.

Users

A user is any person or software agent that will access a GIS server resource. ArcGIS Server maintains a list, called the identity store, of users that are allowed to make resource requests.

Roles

A role is a set of users. Users who make up a role are usually related by function, title, or some other relationship. For example, users who will perform administration of an ArcGIS Server site could be grouped into a role named Administrators, and users who belong to an organization's Human Resources department could be grouped into a role named Human Resources. A user can belong to more than one role. Roles are managed along with users in the identity store.

Permissions

In ArcGIS Server, the authority to access a GIS resource can only be assigned to a role. Individual users can only acquire permissions by inheriting them from their roles. Role-based access control provides the ability to enforce, manage, and audit an organization's access control policies efficiently and effectively. Permissions are managed internally by ArcGIS Server.

Authentication and authorization

To enforce permissions for secured resources, a user is first authenticated, then his or her authorization is verified.

Authentication is the process of verifying the identity of a user. In ArcGIS Server, this can be done by using either ArcGIS token-based authentication or web server authentication.

Authorization is the process of verifying that an authenticated user has the permission to access the requested resource.

Related Topics

[About the identity store](#)

[Managing users in Manager](#)

[Managing roles in Manager](#)

[Restricting access to ArcGIS Server](#)

[Restricting access to GIS web services](#)

[About ArcGIS tokens](#)

About registering your data with the server

[Services](#) » [ArcGIS for Server \(Windows\)](#) » [Publishing services](#) » [How to publish services](#) » [Managing service data](#)

As a server administrator, you have the option to register your data folders, databases, and geodatabases with ArcGIS Server. Data registration gives the server a list of locations that the server administrator has verified that the GIS server can access. Data registration also helps the GIS server understand how to adjust data paths when publishing across machines.

Suppose you're a server administrator and you have a department of GIS analysts that will be publishing services to your server from different machines. Using the system tools in ArcGIS for Desktop, you could register a set of approved folders and databases with the server and communicate these directories to your analysts. By using this method, you can decrease the number of incidents where your analysts encounter permissions problems and be unable to publish to the server.

Data sources you can register with the server

You can register any of the following with ArcGIS:

- Databases accessed through a database connection file (.sde only)
- Enterprise geodatabases
- Folders containing file-based data

You can register any enterprise database compatible with ArcGIS by referencing the database connection file (the .sde file). Databases can contain enterprise geodatabases, and both are accessed through database connection files, so the terms enterprise geodatabase and database are used interchangeably in this topic, unless otherwise specified.

You can also register local and shared operating system folders with the server. These might contain shapefiles, file geodatabases, and other GIS resources. When you register a folder, its subfolders are also registered.

 Note: If your data locations change, you should update your registered data locations using ArcGIS for Desktop or ArcGIS Server Manager.

Before registering data with the server

Registering your databases and data folders does not grant the ArcGIS Server account permissions to access your data. Before registering your data, you'll need to ensure that the ArcGIS Server account has at least read permissions to the data. To learn more about this process, see [Making your data accessible to ArcGIS Server](#).

Installing client software for your database

Before registering a database with ArcGIS Server, you need to ensure that the 64-bit version of the database's client software is installed on each GIS server in your site. For example, if you plan to register a SQL Server database, you must install SQL Server Native Client on each GIS server in your site.

Once you have installed the client software, restart the ArcGIS Server service.

The following links describe the client software needed for each database:

- [Setting up a connection to SQL Server](#)

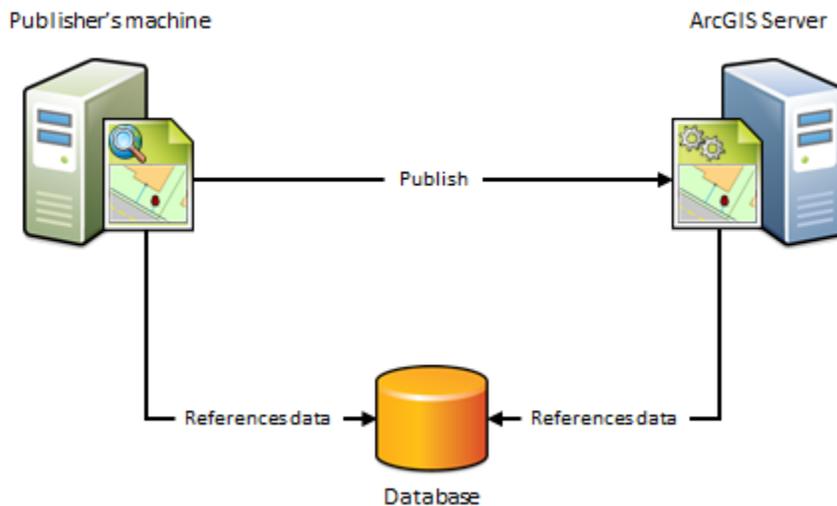
- [Setting up a connection to Oracle](#)
- [Setting up a connection to PostgreSQL](#)
- [Setting up a connection to Informix](#)
- [Setting up a connection to DB2](#)

Scenarios for registering your data with ArcGIS Server

Before registering your data, examine the following scenarios and consider how your workflows relate:

If the publisher's machine and the server are working with the same database

If the publisher's machine and the server are working with the same database, import the publisher's database connection and set the server's database connection to Same as publisher's connection when registering your data.



When to use this scenario

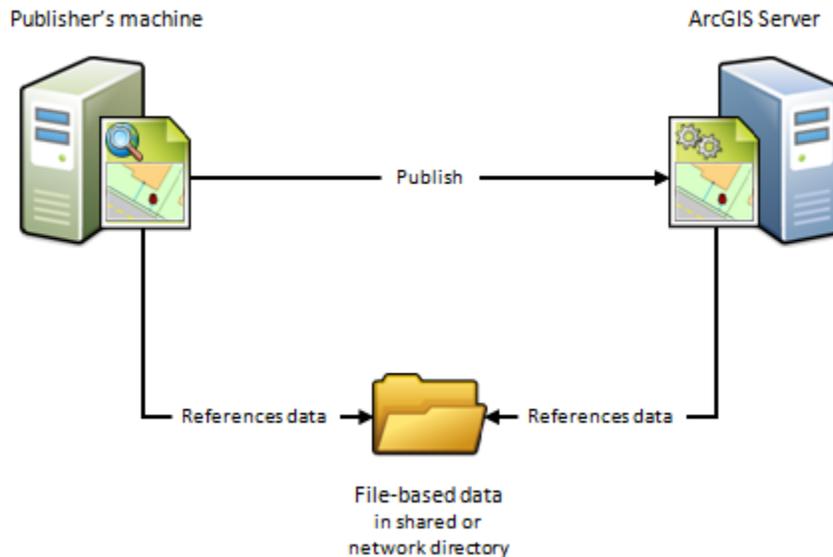
Use this scenario if you want to avoid having a copy of the data placed on the server. For example, suppose you want to publish a map service to ArcGIS Server using data from an on-premises enterprise geodatabase. To avoid having a copy of the data referenced by your map document placed on the server, import the publisher's database connection and set the server's database connection to Same as publisher's connection. After you publish, the map document continues to reference the data stored in your enterprise geodatabase.

When not to use this scenario

- If your data resides in a file geodatabase or file directory. Instead, use the [next](#) scenario.
- If you want to maintain a separate copy of the data in your enterprise geodatabase for web use.

If the publisher's machine and the server are working out of the same folder

If the publisher's machine and the server are working out of the same folder, specify the publisher's folder path and set the server's folder path to Same as publisher's path when registering your data. This scenario is just like the previous one, except it uses folders, not geodatabases.



When to use this scenario

Use this scenario if you want to avoid having a copy of the data placed on the server. For example, suppose you want to publish a geoprocessing service to ArcGIS Server using data from a network directory. To avoid having a copy of the geoprocessing service's data placed on the server, specify the publisher's folder path and set the server's folder path to Same as publisher's path. After you publish, the geoprocessing service continues to reference the geoprocessing model, inputs, outputs, scripts, and project data stored in your network directory.

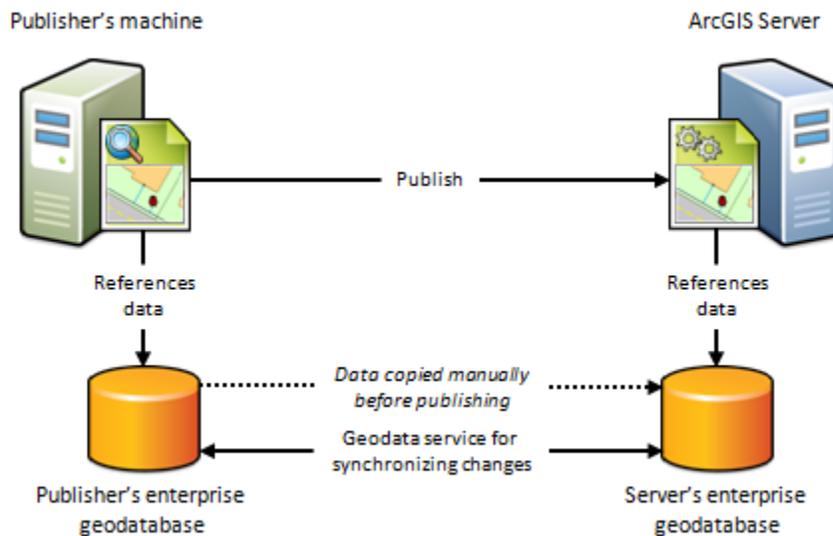
This scenario is also beneficial if you have a Linux-based ArcGIS Server site that manages all of your data and you've set up Samba to allow file sharing between Windows and Linux. For example, if you want to publish a map document that references the data on your Linux machine, register the Samba directory (\\net\data) as the publisher's folder and register the Linux directory (/net/data) as the server's folder. When you publish, the map document is automatically modified to reference the directory on the Linux machine.

When not to use this scenario

- If your data resides in a database. Instead, use the [preceding](#) scenario.
- If you want to publish feature or WFS-T services.

If the publisher's machine and the server are working with different databases

Because of firewalls, differences between computing platforms, or the desire to keep a separate copy of the data for web use, the publisher and the server may each be working with different databases. To register your data using this scenario, you'll need to import both the connection to the publisher's database and the connection to the server's database.



When to use this scenario

Use this scenario if you want to maintain a separate copy of the data in your on-premises enterprise geodatabase for web use. In this case, you're responsible for making sure a copy of the data in your publisher's geodatabase exists in the server's geodatabase. This scenario can only be used with enterprise geodatabases; not databases.

One way to get your data into the server's enterprise geodatabase is to check [Create geodata service for server database](#) when registering your enterprise geodatabases. Selecting this option automatically creates a [geodata service](#) that you can use to manually send a replica of the data in the publisher's geodatabase to the server's geodatabase.

You can also use the geodata service to synchronize the enterprise geodatabases, thereby ensuring that any subsequent changes made to the publisher's database are reflected in the server's database. This is particularly advantageous in cloud deployments, such as ArcGIS Server on Amazon Web Services, as it does not require someone to log in to the cloud machine and arrange for the data transfer.

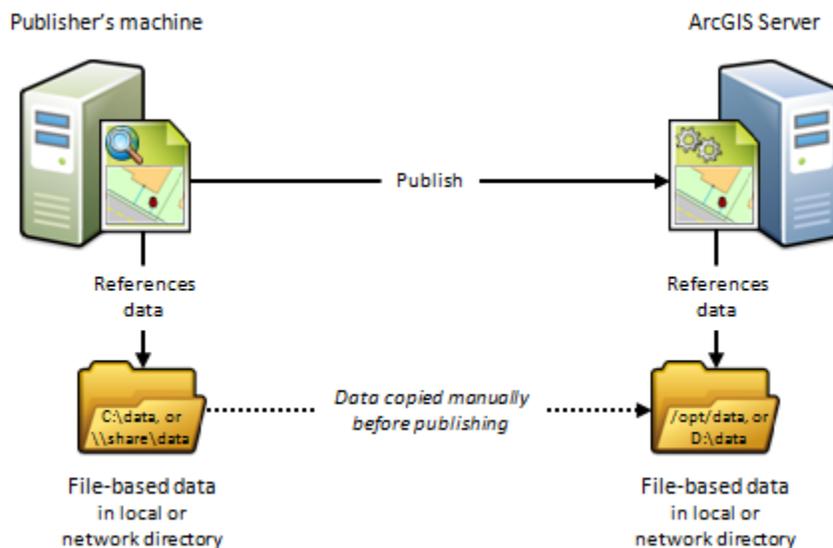
This scenario is also well suited for publishing feature services to on-premises or cloud servers. For example, if you publish a feature service using this scenario, edits made on-premises could be pushed to the server's geodatabase, thereby becoming available to end users of your feature service. Conversely, if web editors change any features in the server's geodatabase, the edits can be synchronized with the publisher's geodatabase.

When not to use this scenario

- If your data resides in a file geodatabase or file directory. Instead, use the [next](#) scenario.
- If your data resides in a database. Instead use the first scenario.
- If you do not want to maintain a separate copy of your geodatabase on the server.

If the publisher's machine and the server are working out of different folders

Because of firewalls, differences between computing platforms, or the desire to keep a separate copy of the data for web use, the publisher and the server may each be working with their own data folder. To register your data using this scenario, you'll need to enter the path to both the publisher's folder and the server's folder.



When to use this scenario

This scenario is useful for Linux deployments, cloud deployments, or any deployment where you want publishers and web users to work with separate copies of the data.

For example, if you want to publish a map service from ArcGIS for Desktop to a Linux-based ArcGIS Server site, you could create an identical copy of your map document's data and place the data on the Linux-based server. After you register both directories with the server and publish, the map document is automatically modified to reference the folder on the Linux-based server.

This scenario is beneficial if you are publishing to a cloud-based server such as ArcGIS Server on Amazon Web Services. For example, you can copy your on-premises data and place it in any directory you want to in the cloud. When you publish, the data paths are automatically modified to reference the directory on the cloud server. The disadvantage of this approach is that it requires someone to log in to the cloud

machine and arrange for the data transfer to the cloud (which could be performed through FTP, remote desktop copy and paste, and so on).

When not to use this scenario

- If your data resides in an enterprise geodatabase. Instead, use the [preceding](#) scenario.
- If your data resides in a database. Instead, use the first scenario.
- If you do not want to maintain a separate copy of your data on the server.

How to register your data with the server

You can register your data folders and enterprise databases with the server using ArcGIS Server Manager or ArcGIS for Desktop. For full instructions, see the following:

- [Registering your data with ArcGIS Server using Manager](#)
- [Registering your data with ArcGIS Server using ArcGIS for Desktop](#)

Related Topics

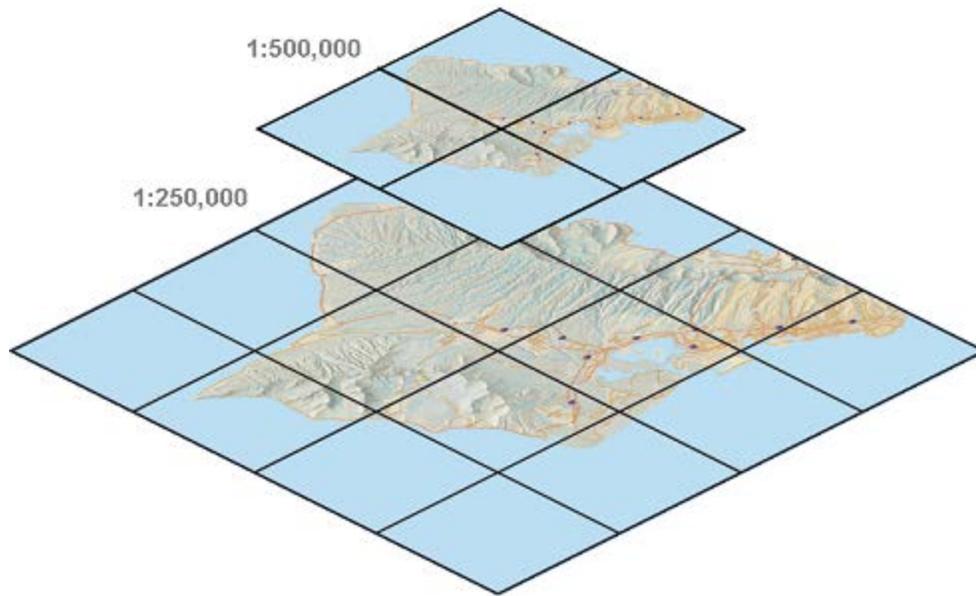
[Making your data accessible to ArcGIS Server](#)

[Copying data to the server automatically when publishing](#)

What is map caching?

[Services](#) » [ArcGIS for Server \(Windows\)](#) » [Publishing services](#) » [Caching services](#)

Map caching is a very effective way to make your map and image services run faster. When you create a map cache, the server draws the entire map at several different scales and stores copies of the map images. The server can then distribute these images whenever someone asks for a map. It's much quicker for the server to return a cached image than to draw the map each time someone requests it. Another benefit of caching is that the amount of detail in the image doesn't noticeably affect how quickly the server can distribute the copy.



What happens during the caching process?

Caching does not happen automatically. To make a cache, you first need to design the map and share it as a service. Then, you'll set some cache properties and begin creating tiles. You can choose to create all the tiles at once or allow some of the tiles to be created on demand, that is, when someone first visits them.

When you cache a map, you draw it at more than one scale so that users can zoom in and out of the map. When choosing properties for your cache, you'll need to determine what scales you want to use for caching. If you're just trying out the caching tools for the first time, you can let the computer pick some scales for you. However, you should usually pick your scales earlier so that you can design your map to look good at those scales. Write down the scales and use them as you do your cartographic design in ArcMap. When it's time to create the cache, you can enter the scales into the cache generation tool.

There are other properties that are important to understand when you create the cache. You can find out more by reading [Available map cache properties](#).

The cache is stored in your server cache directory. When you installed ArcGIS for Server, a server cache directory was created for you in a local folder. If you add more GIS servers to your site, you need to share your server cache directory (using operating system options such as Windows file sharing) so it can be accessed by other machines in your site.

The scales that you pick and the properties you set for the cache are the tiling scheme. Each cache has a tiling scheme file that you can import when you create new caches so that all your caches use the same tile sizes and scales. This can help the performance of your web applications that contain more than one cached service. Optionally, you can choose to use the well-known tiling scheme of ArcGIS Online, Google Maps, and Bing Maps so that you can easily overlay your caches with these online mapping services.

Can I cache all my maps?

A map cache represents a snapshot of your map at one point in time. Because of this, caches work best with maps that do not change frequently. These include street maps, imagery, and terrain maps.

If your data tends to change, you might still be able to use the caching tools to periodically update the cache. You can even schedule these updates to occur automatically. To understand whether your frequently changing map can still be cached, it helps to ask these questions:

How up-to-date does my map need to be?

If the data you see on the map needs to be live, with no time delay acceptable, caching is not appropriate. However, if a short delay is acceptable and the cache updates can be performed within that time window, you can still use caching.

How big is the cache and how widespread are the changes to my data?

These two questions go together. A large cache takes more time to create. It may be practical to update your large cache only if you can isolate the changed areas and update those alone. If the cache is small, you might be able to quickly rebuild the entire cache.

If the update cannot keep up with the changes in an acceptable amount of time, the map is not appropriate for caching.

After considering the questions above, use caching whenever it's appropriate. The performance benefit that you gain is probably worth the investment it takes to create and update the cache.

To learn more about updating your cache, see [Map cache updates](#).

Can I still get to the underlying data?

Although map caches represent pictures of the data, you can still allow others to perform identify, search, and query operations on your map service. These tools can get the geographic locations of features from the server and return the results. The application draws the results in its native graphics layer format on top of the cached image.

How do I get started with caching?

To get started with caching, you must first publish a map or image service. As part of the publishing process, you will set properties in the Service Editor dialog box. This is where you can define the scales and extent of your cache. Follow the steps in [How to publish a service](#) to learn how to reach the Service Editor.

You can choose to create the tiles immediately when you publish the service (appropriate for small caches) or build the cache on your own after publishing (appropriate for large caches where you want to geographically limit the amount of cache built at large scales). If you build the cache on your own, you will use the [Manage Map Server Cache Tiles](#) geoprocessing tool, available in the Server Tools toolbox.

Enabling SSL using a self-signed certificate

[Services](#) » [ArcGIS for Server \(Windows\)](#) » [Administering ArcGIS for Server](#) » [Securing your ArcGIS Server site](#) » [Configuring ArcGIS Server to be secure](#)

This tutorial shows you how you can enable SSL for ArcGIS Server using a self-signed certificate. The steps to enable SSL using a self-signed certificate are:

Steps:

1. [Create a new self-signed certificate.](#)
2. [Configure ArcGIS Server to use the certificate.](#)
3. [Configure each GIS server in your deployment.](#)
4. [Enable SSL for your site.](#)
5. [Access your site using SSL.](#)

Create a new self-signed certificate

Steps:

1. Log in to the ArcGIS Server Administrator Directory: `http://[host name]:6080/arcgis/admin`.
2. Navigate to machines > [machine name] > sslcertificates.
3. Click generate.
4. Enter values for the parameters on this page:

Option	Description
Alias	A unique name that easily identifies the certificate.
Key Algorithm	Use RSA (the default) or DSA.
Key Size	Specifies the size in bits to use when generating the cryptographic keys used to create the certificate. The larger the key size, the harder it is to break the encryption; however, the time to decrypt encrypted data increases with key size. For DSA, the key size can be between 512 and 1,024. For RSA, the recommended key size is 2,048 or greater.
Signature Algorithm	Use the default (SHA1withRSA). If your organization has specific security restrictions, then one of the following algorithms can be used: SHA256withRSA, SHA384withRSA, SHA512withRSA, SHA1withDSA for DSA.

Use the domain name of your server name as the common name.

Common Name If your server will be accessed on the Internet through the URL <https://www.myarcgis.com:6443/arcgis/>, use www.myarcgis.com as the common name.

If your server will only be accessible on your local area network (LAN) through the URL <https://hostname:6443/arcgis/>, use hostname as the common name.

Organizational Unit The name of your organizational unit, for example, GIS Department

Organization The name of your organization, for example, Esri

City or Locality The name of the city or locality, for example, Redlands

State or Province The full name of your state or province, for example, California

Country Code The abbreviated code for your country, for example, US

Validity The total time in days during which this certificate will be valid, for example, 365.

5. Click Generate to generate the certificate.

Configure ArcGIS Server to use the SSL certificate

To specify the SSL certificate that ArcGIS Server should use:

Steps:

1. Log in to the ArcGIS Server Administrator Directory: [http://\[host name\]:6080/arcgis/admin](http://[host name]:6080/arcgis/admin).
2. Navigate to machines > [machine name].
3. Click edit.
4. Enter the name of the SSL certificate that you want to use in the box for Web server SSL Certificate.
5. Click Save Edits to apply your change.
6. On the current page, view the property Web server SSL Certificate to verify that the desired SSL certificate will be used for SSL.

Configure each GIS server in your deployment

If you have a multimachine deployment of ArcGIS Server, you must create a new self-signed certificate for each GIS server that participates in your site and configure each machine to use the certificate.

Enable SSL for your site

Steps:

1. Log in to the ArcGIS Server Administrator Directory: [http://\[host name\]:6080/arcgis/admin](http://[host name]:6080/arcgis/admin).
2. Navigate to security > config > update.
3. For the Protocol parameter, choose the HTTP and HTTPS option then click Update. This will automatically restart your ArcGIS Server site.
4. After your site is restarted, verify that you are able to access the URL [https://\[host name\]:6443/arcgis/admin](https://[host name]:6443/arcgis/admin). If you do not get a response from this URL, ArcGIS Server was unable to use the specified SSL certificate. Check your SSL certificate and configure ArcGIS Server to use a new or different SSL certificate.
5. If you are able to access the URL [https://\[host name\]:6443/arcgis/admin](https://[host name]:6443/arcgis/admin), navigate to security > config > update.
6. For the Protocol parameter, choose the HTTPS Only option then click Update.

Access your site using SSL

Once SSL has been configured, ArcGIS Server listens on port 6443 for HTTPS requests. Use the URLs below to securely access ArcGIS Server:

ArcGIS Server Manager [https://\[server name\]:6443/arcgis/manager](https://[server name]:6443/arcgis/manager)

ArcGIS Server Services Directory [https://\[server name\]:6443/arcgis/rest/services](https://[server name]:6443/arcgis/rest/services)