Enterprise-scale Farms for SharePoint Server 2013

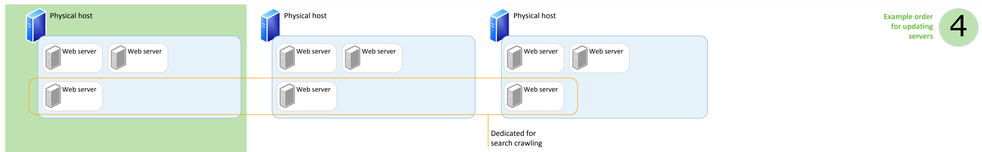
EXAMPLE ENTERPRISE-SCALE FARM

The architecture represented in this model is an example enterprise-scale farm based on a customer with 90,000 users and 100 million search items. This architecture can be used as a reference point for planning an enterprise-scale farm. Actual numbers of servers and components depends on many variables such as usage, services, size of documents, rates of change and freshness requirements for search results, and many others.

Web servers

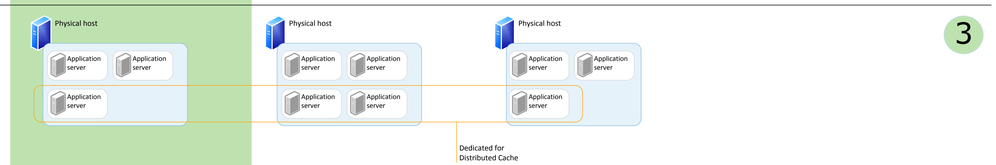
A Web server typically supports 10,000-20,000 users. For 90,000 users this architecture starts with six Web servers to serve user requests and leaves room for additional Web servers, if needed.

Two-three Web servers that are dedicated for search crawling is a good starting point, depending on rates of change and freshness requirements.



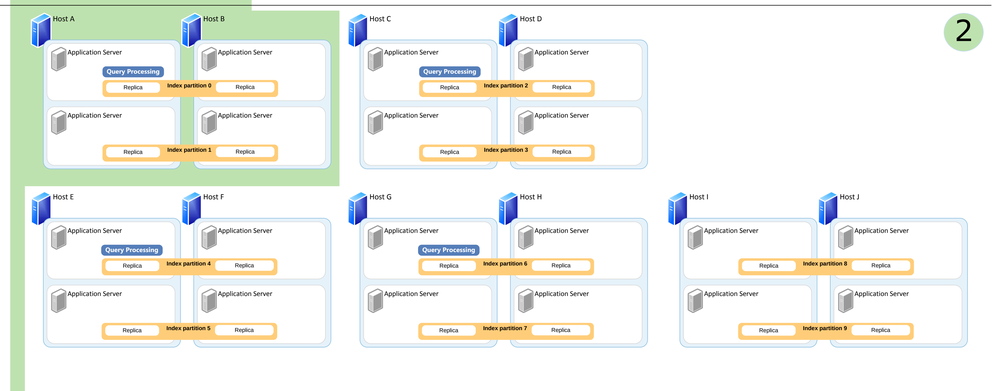
Application Servers — General

- Four servers dedicated for distributed cache
- Central Administration
- Access Services
- App Management
- Business Data Connectivity
- Excel Services
- Machine Translation Service
- Managed Metadata
- PerformancePoint
- PowerPoint Conversion
- Secure Store Service
- State Service
- Usage and Health Data Collection
- User Profile
- Visio Graphics Service
- Word Automation Services
- Work Management
- Workflow

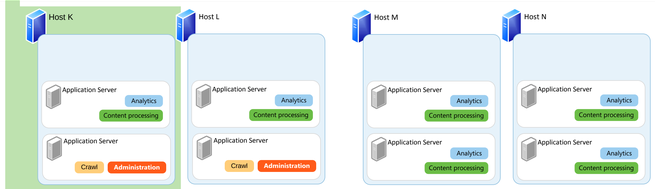


Search application servers — Index and query processing components

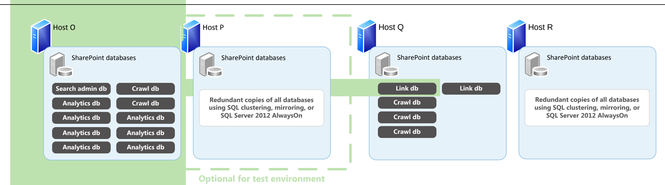
Search architecture to support 100 million items



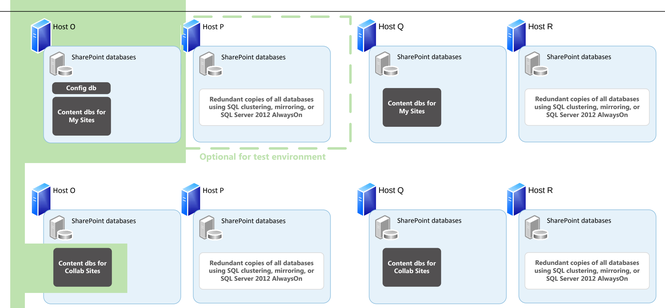
Search application servers — All other search application components



Databases — Search

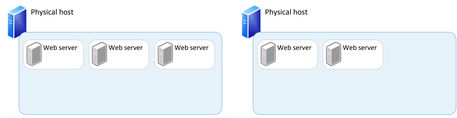


Databases — Content



OFFICE WEB APPS SERVERS

Office Web Apps Server is not part of the SharePoint 2013 farm. The following architecture provides an example of an appropriately-scaled Office Web Apps Server environment to accompany the enterprise-scale SharePoint farm that is represented in this model.



PATCHING AND UPDATING ENTERPRISE-SCALE FARMS

General guidance for SharePoint environments

Operating system updates

The process that updates the operating system for servers in a SharePoint farm is uncomplicated and can be performed server by server (one server at a time):

- Remove a server from the rotation of the load balancer.
- Update the server.
- Return the server to the rotation of the server farm.

This guidance applies to both physical and virtual servers.

SQL Server updates

Whether you can apply updates of SQL Server in a high availability state depends on the implementation of SQL Server:

- SQL mirroring — You can update servers in a high availability state.
- SQL clustering — The server farm that hosts SharePoint will be down while you update the config db.

See SQL Server documentation for guidance about how to patch SQL Server, including guidance for SQL Server 2012 AlwaysOn.

Office Web Apps Server

You can update Office Web Apps Server independent of updates of the SharePoint environment.

Monitoring servers

Be sure to coordinate with your operations team when servers are taken offline for maintenance. Remote actions that bring a server back online during the patching process can produce unrecoverable error states.

Disaster recovery environments

If a standby disaster recovery environment is in place, be aware of the effect of using SQL log shipping or SQL Server 2012 AlwaysOn features when the farms are on different versions of SQL Server or SharePoint Server.

Applying SharePoint updates to a large server farm

Safest update method

The safest method to update a SharePoint farm is to take the entire farm offline, update all servers, and then bring the farm back online. It is important to test the update process in a test environment even when you use this method. This method requires a maintenance window that might not be practical for all organizations.

High availability updating

High availability updating involves more planning, testing, and coordination. The general outline for the process includes the following steps.

1. Update servers by role, in the following order:

- **Application servers** — Start with the services that are most important to the organization. The example to the left updates search servers before other application servers. Within the search service application roles, servers that host the index are updated after servers that host other search roles. If a different service is more important to your organization, such as the Business Data Connectivity Service, update these servers first.
- **Web servers**
- **Database servers** — The example to the left updates content databases before search databases.

2. For each role, update half the servers at a time.

- Remove the first half of the servers from the rotation of the load balancer. Update these servers.
- Before you return updated servers to the rotation, remove the second half of the servers of the same role from the load balancing rotation.
- Return the first half of the servers to the load-balancing rotation.
- Update the second half of the servers and then return these to the load-balancing rotation.

3. Run the Psconfig command-line tool to update SharePoint databases.

If you updat half of the servers of the same role at the same time, you eliminate the possibility that a given role will run on the same farm with two different versions. Site will not be accessible while Psconfig runs.

Important — Be sure to test the upgrade process and code in a test environment before you attempt the update process in production.

Using database snapshots during the update process

A new feature of SharePoint 2013 enables you to update databases in a production environment. The snapshot method takes a snapshot of the current database and then performs all update operations that apply to the database, and optionally to its contents.

The existing connections to the content database are set to use the snapshot in read-only mode for the duration of the update, and then switched back after successful completion of upgrade. A failed upgrade reverts the database to its state when the snapshot was taken.

This feature is implemented by using the Windows PowerShell update-spcontentdatabase cmdlet and only works for versions of SQL Server that support creation and use of snapshots, such as, SQL Server Enterprise edition.

Updating multiple farms

When you update environments with multiple farms, update farms in the following order:

- Service farms
- My site farm
- Content farms

It's important to update service farms before you update farms that consume services. Service farms support connections to farms that do not have the same updates installed. However, if you update a consuming farm before you update a service farm, the consuming farm might encounter issues when it connects to the service farm.

Working around sites with issues

If specific sites cause upgrade issues, you will encounter error messages with each server that you update. The best course of action is to resolve the issues with the sites. However, if you cannot resolve issues during the update timeframe, you can consider the following solutions:

1. Move the sites to a dedicated database.
2. Remove the database from the farm during the update process.
3. Reattach the database after the update process.

Scaling to a multi-farm environment for manageability

This model provides an example of a very large server farm. While farms of this size are supported, you could more easily manage the environment if you split it into two farms. Given the search requirements for this environment, the recommendation for manageability is to create a separate search farm that hosts all search roles. The resulting content farm can host all other service applications. Another option is to divide the environment into three farms, and the non-search service applications reside on a dedicated services farm.

For more information about how to design dedicated search farms see the **Enterprise Search Architectures for SharePoint Server 2013** model.

If you divide the environment into two or more farms, you greatly reduce the time that is required to update any one of the farms.

Top mistakes to avoid when you update farms

- **Monitoring servers during the update process** — An operations team can initiate remote actions that interfere with servers that are offline for updating. Coordinate with the operations team to remove servers from monitoring.
- **Updating one server at a time** — If you return updated servers to a farm one-by-one, you can cause server roles of two different versions to run in the production farm. This can compromise performance or cause error states. Instead, for each server role, take half of the servers offline and update these servers. Before you return these servers to the rotation, remove the second half of the servers that run the same role from the rotation.
- **Waiting to update databases** — Databases do not always need to be updated at the same time as the rest of the farm. This allows critical updates to be applied to a production farm with minimal downtime. However, databases should be updated within the next week or month.
- **Not testing the update process and code** — See below.

Build and use a test environment for SharePoint updates

Build the environment

It is important to replicate a portion of the production environment in a test environment to test the update process and code. The servers highlighted with the green background represent the maximum number of servers that you need to test the update process for a farm of this size. This environment consists of 8-10 servers, depending on whether the redundant copies of the specified database servers are included or not. Be sure to include the Link db and content databases for both My Sites and team sites in the test environment (also highlighted in green). These can be shared on a VM with other dbs.

The advantages of building a larger test environment include the following:

- The environment more closely resembles the production environment with combination of virtual roles shared on physical hardware.
- The performance of updating the servers will more closely match the performance in production. This will give you an idea of how long it will take to update the production environment.
- You can use procedures and processes that you develop in the test environment when you update the production environment.

Smaller environments can be used to test the update process. For example, the following three-server environment represents the smallest environment that is recommended for an enterprise-scale farm:

- Host 1 — 2 Web servers and 2 application servers
- Host 2 — All search roles on one server
- Host 3 — SQL Server with all database roles included

In any test environment, be sure to include at least two instances of every role represented on the farm.

Testing the update process

The primary purposes for testing include:

- Confirm that the updates perform as intended and do not cause issues for the server farm. This is especially important if you apply more than one update at the same time.
- Identify "breaking changes" that might occur with a specific update or combination of updates.
- Test the update process for your organization.

The recommended process for testing validates both the updates and the process that applies updates for your organization:

- Test and document the process and procedures, including the specific order of roles that are updated.
- Validate the health of the server farm.
- Start all over with a clean environment and use the documentation that you produced to reapply the updates.

When updates are applied in the production environment, replicate the same process and procedures.

Microsoft®