

# SharePoint 2010 Products: Deployment

## Requirements

Before you can deploy Microsoft® SharePoint® Server 2010 or Microsoft® SharePoint® Foundation 2010, you must meet the following requirements:

### Hardware requirement: 64-bit

SharePoint Server 2010 and SharePoint Foundation 2010 are 64-bit applications and can only run on a 64-bit edition of the Windows Server® 2008 operating system. You must have hardware that supports the use of a 64-bit operating system and 64-bit SQL Server.

### Operating system requirement: Windows Server 2008 or Windows Server 2008 R2

SharePoint Server 2010 and SharePoint Foundation 2010 must be run on a 64-bit edition of Windows Server 2008 with Service Pack 2 or Windows Server 2008 R2.

### Database requirement: 64-bit SQL Server 2005 SP2 or 64-bit SQL Server 2008

For server farm installations of SharePoint Server 2010 and SharePoint Foundation 2010, you must be running 64-bit versions of Microsoft SQL Server® 2005 or Microsoft SQL Server 2008 (each with the appropriate service packs and updates) on your database servers.

For more information, see the article [Determine hardware and software requirements](#) on TechNet.

## Related models

This model covers planning for deployment of SharePoint Server 2010 or SharePoint Foundation 2010. For further information about planning your infrastructure for these products, see the following models:

- Topologies for SharePoint Server 2010
- Services in SharePoint 2010 Products
- Cross-farm Services in SharePoint 2010 Products
- Hosting Environments in SharePoint 2010 Products
- Upgrade (1-4)

## Best practices for maintaining multiple environments

### Keep your environments synchronized by using the following:

- Profile replication engine – a tool for keeping profile and social data synchronized across farms.
- Content deployment – a method of moving content between authoring, staging, and live environments.
- Mirroring and log shipping – two techniques for keeping content changes synchronized across farms.

### Keep your environments clean by using the following best practices:

Be sure to reformat your computers before re-using hardware between environments or within an environment. Do not simply uninstall and reinstall. If there are old customizations or configurations, they can affect how that computer works and introduce errors into your environment.

## Deployment stages and environments

Over the course of planning, developing, testing, and rolling out SharePoint 2010 Products, you perform several deployments, such as an initial proof of concept, a pilot or pre-production environment, and your production environment.

If you are using the same hardware to move through these phases, be sure to reformat each computer between each phase. The number of environments below might seem intimidating, but note that some are temporary, and many can be virtualized.

Always use development best practices for managing your solutions throughout their lifecycle. For more information, see the [Application Lifecycle Management Resource Center](#) on MSDN.

## Deployment stage and environment

### Planning

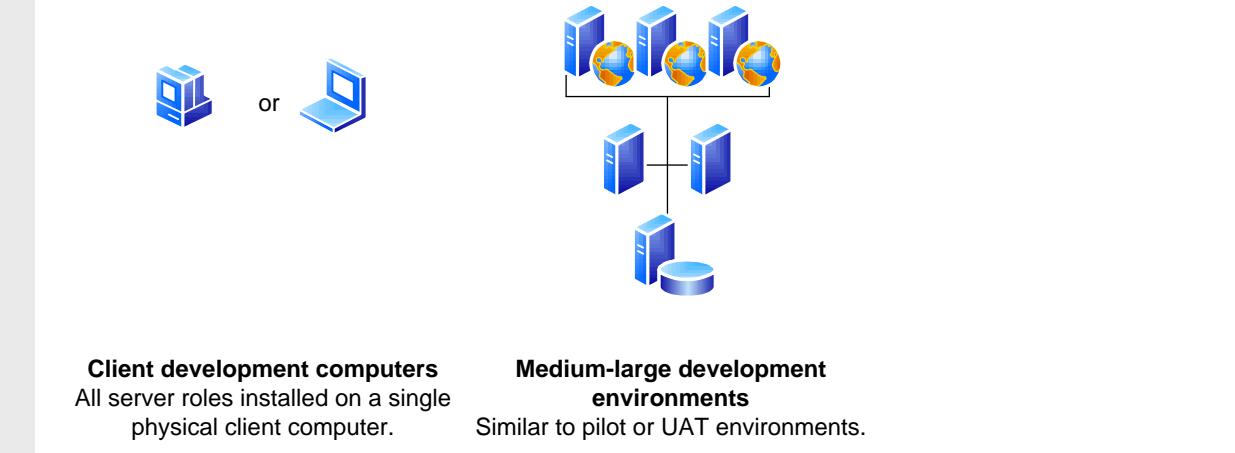
Purpose	Steps	Output
Before you can deploy, you must plan the solution you want to deploy. After the planning stage, you move through the deployment stages in the following table, updating and revising your plans as you test.	<ul style="list-style-type: none"> <li>• Perform business analysis</li> <li>• Determine goals and objectives</li> <li>• Determine infrastructure requirements</li> </ul>	<ul style="list-style-type: none"> <li>• Solution plan</li> <li>• Topology and resource requirements</li> </ul>

### Development

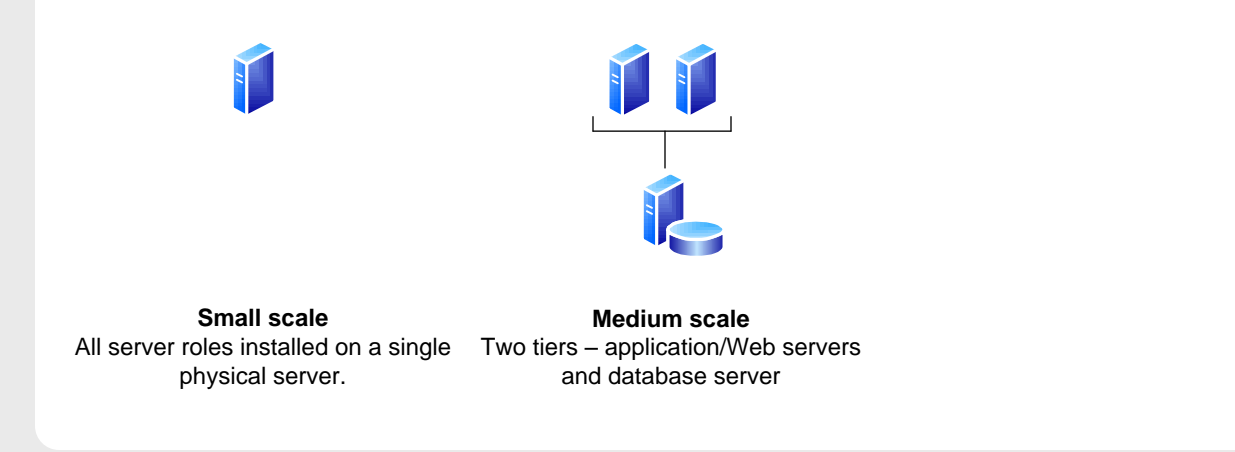
Purpose	Steps	Output	Characteristics	Expected duration and stability
Used for developing applications and solutions for SharePoint 2010 Products.	<ul style="list-style-type: none"> <li>• Deploy development computers or farm</li> <li>• Develop solution</li> <li>• Test and evaluate solution</li> <li>• Refine solution</li> </ul>	<ul style="list-style-type: none"> <li>• Solution</li> </ul>	<p>Development environments exist throughout the lifecycle of the project. Initial development of the solution is followed by testing and refining the solution.</p> <p>Note that SharePoint Workspaces can be used to keep development environments synchronized.</p>	<ul style="list-style-type: none"> <li>• Exists throughout solution lifecycle</li> <li>• Not a stable environment – no service-level agreement for users</li> </ul>

## Example topologies

Topologies vary, ranging from solution development computers running Windows 7 to large scale pilot and UAT environments.



Topology is usually a single server or small farm.



### Proof of concept (POC)

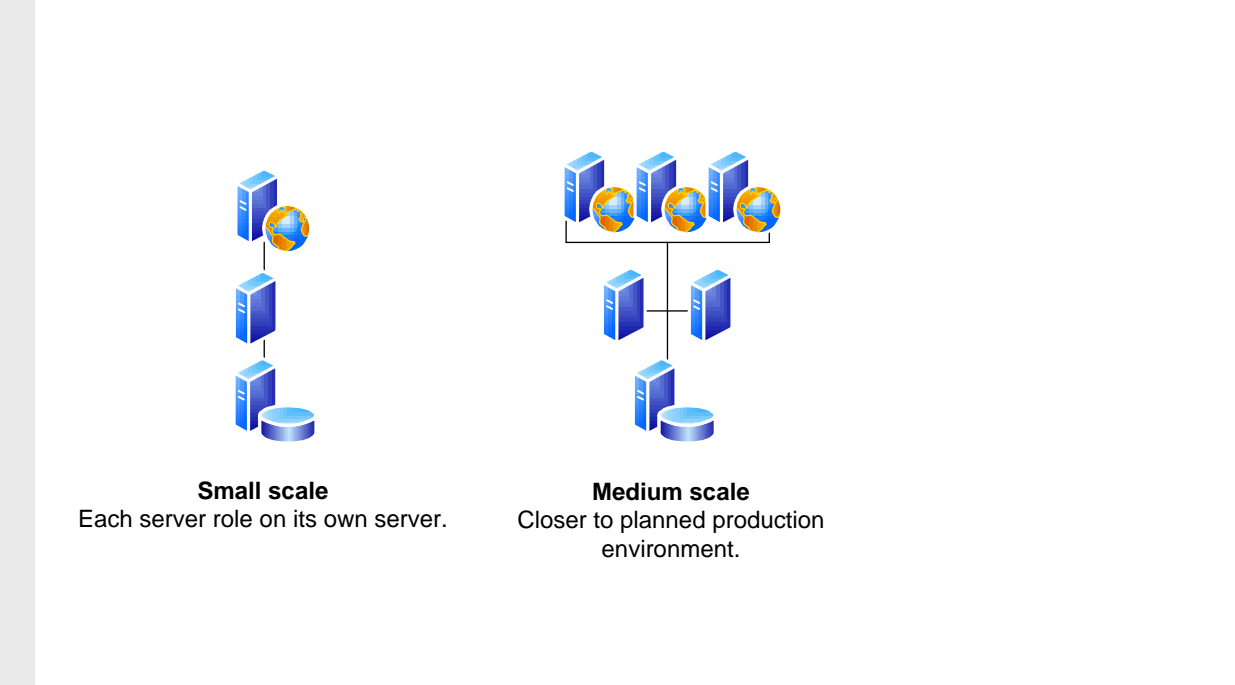
Purpose	Steps	Output	Characteristics	Expected duration and stability
Used for determining whether a solution will meet business needs and to determine an appropriate infrastructure plan.	<ul style="list-style-type: none"> <li>• Deploy farm</li> <li>• Deploy solution</li> <li>• Collect benchmark data</li> <li>• Evaluate proof of concept</li> <li>• Refine goals and infrastructure requirements</li> </ul>	<ul style="list-style-type: none"> <li>• Updated solution plan</li> <li>• Updated topology and resource requirements</li> </ul>	<p>POCs are often created during evaluation phase to test the product or solution. A POC can be hosted on a development environment or on a small scale production computer. Sometimes the POC moves through several stages in continuous development.</p>	<ul style="list-style-type: none"> <li>• Set up, reviewed, and then discarded</li> <li>• Not a stable environment – no service-level agreement for users</li> </ul>

### Pilot (small scale test)

Purpose	Steps	Output	Characteristics	Expected duration and stability
Used to test a solution on a small scale. Pilots are used to test solution readiness (no real data, just functionality testing). A pilot can also be used to test for production characteristics (real data, actual work); this is recommended.	<ul style="list-style-type: none"> <li>• Deploy pilot farm</li> <li>• Deploy pilot solution</li> <li>• Collect benchmark data</li> <li>• Evaluate pilot</li> <li>• Refine goals and infrastructure requirements</li> <li>• Determine operations plan</li> </ul>	<ul style="list-style-type: none"> <li>• Updated solution plan</li> <li>• Updated topology and resource requirements</li> <li>• Operations plan</li> </ul>	<p>A limited set of users has access to the environment to test the functionality and performance of the solution and infrastructure.</p>	<ul style="list-style-type: none"> <li>• Limited time frame</li> <li>• Limited service-level agreement for solution readiness testing</li> <li>• Production service-level agreement for production testing</li> </ul>

- Tips for a successful pilot project**
- Get executive sponsorship for the project.
  - Establish a clear schedule.
  - Make the environment production-class, but smaller scale.
  - Limit the pilot testing to no more than a few hundred users.
  - Establish an appropriate service-level agreement.
  - Treat the pilot as business-critical, not as a lab environment. In other words, if the environment were to go offline, it would stop work for the pilot users.
  - Test operational tasks (backup and recovery, health monitoring) as well as solution elements.
  - Perform surveys of the users to get feedback on the solution and the operation of the environment.

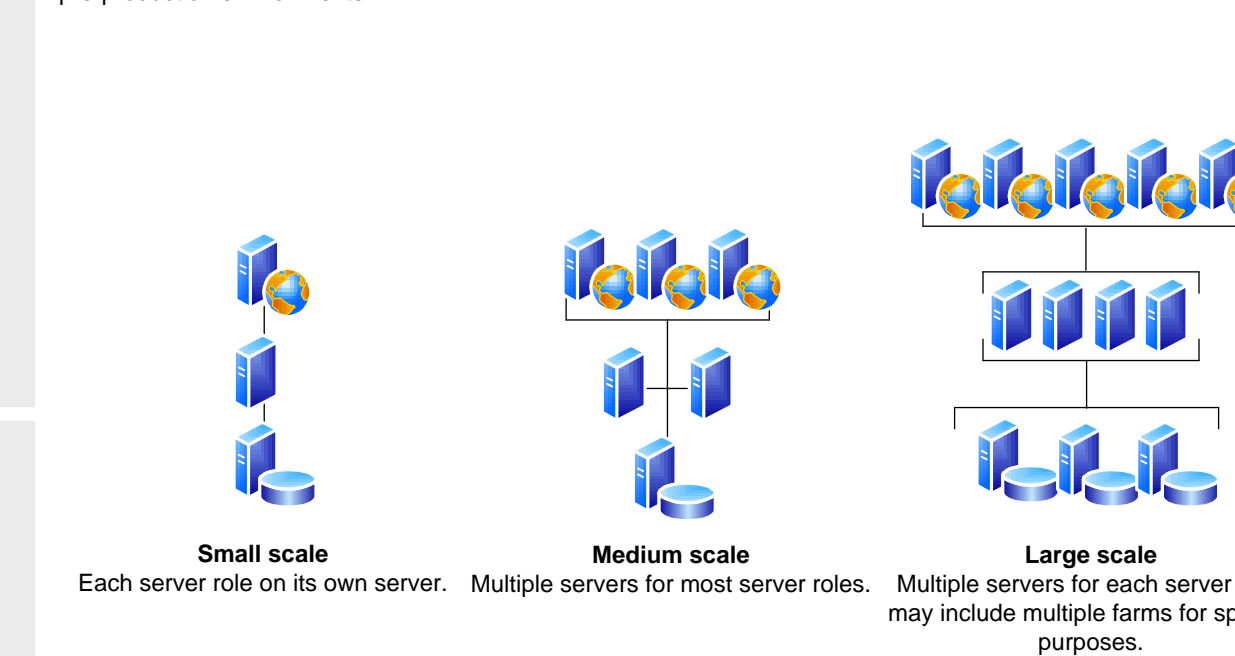
Topologies vary, depending on the type of solution, the number of pilot users, and the type of testing desired. Pilot environments are generally scaled down, but representative versions of the planned production environment.



### User acceptance test (UAT)

Purpose	Steps	Output	Characteristics	Expected duration and stability
A pre-production environment used for testing solutions against a subset or complete copy of production data. Also used for validating the backups and operational procedures.	<ul style="list-style-type: none"> <li>• Deploy UAT farm</li> <li>• Deploy UAT solution</li> <li>• Implement operations plan</li> <li>• Evaluate solution</li> <li>• Evaluate operations plan</li> <li>• Test for capacity and performance</li> </ul>	<ul style="list-style-type: none"> <li>• Updated operations plan</li> </ul>	<p>Topology should be as similar to production environment as possible.</p> <p>Testers ensure that all solution elements function as expected in network and security conditions that match the conditions of the production environment.</p>	<ul style="list-style-type: none"> <li>• Long-term availability – this is a stable pre-production environment</li> <li>• Limited service-level agreement for users</li> </ul>

Topologies vary, depending on the business needs. For UAT testing, it is recommended that they be near the scale of the production environment. If load testing is the goal, then the hardware should be identical. For production environments, topologies vary depending on the solution and performance characteristics that were validated in the pre-production environments.



### Production

Purpose	Steps	Output	Characteristics	Expected duration and stability
This is the live environment that your users interact with.	<ul style="list-style-type: none"> <li>• Deploy production farm</li> <li>• Deploy production solution</li> <li>• Implement operations plan</li> <li>• Deploy additional environments: <ul style="list-style-type: none"> <li>• Authoring and staging farms</li> <li>• Geo-distributed farms</li> <li>• Services farms (Search, Taxonomy)</li> </ul> </li> </ul>	<ul style="list-style-type: none"> <li>• Deployed production farms</li> </ul>	<p>It is business critical and appropriate service-level agreements are in place.</p>	<ul style="list-style-type: none"> <li>• Long-term availability – this should be a stable environment</li> <li>• Full service-level agreement for users, appropriate to the solution and business requirements</li> </ul>

## Installation and configuration

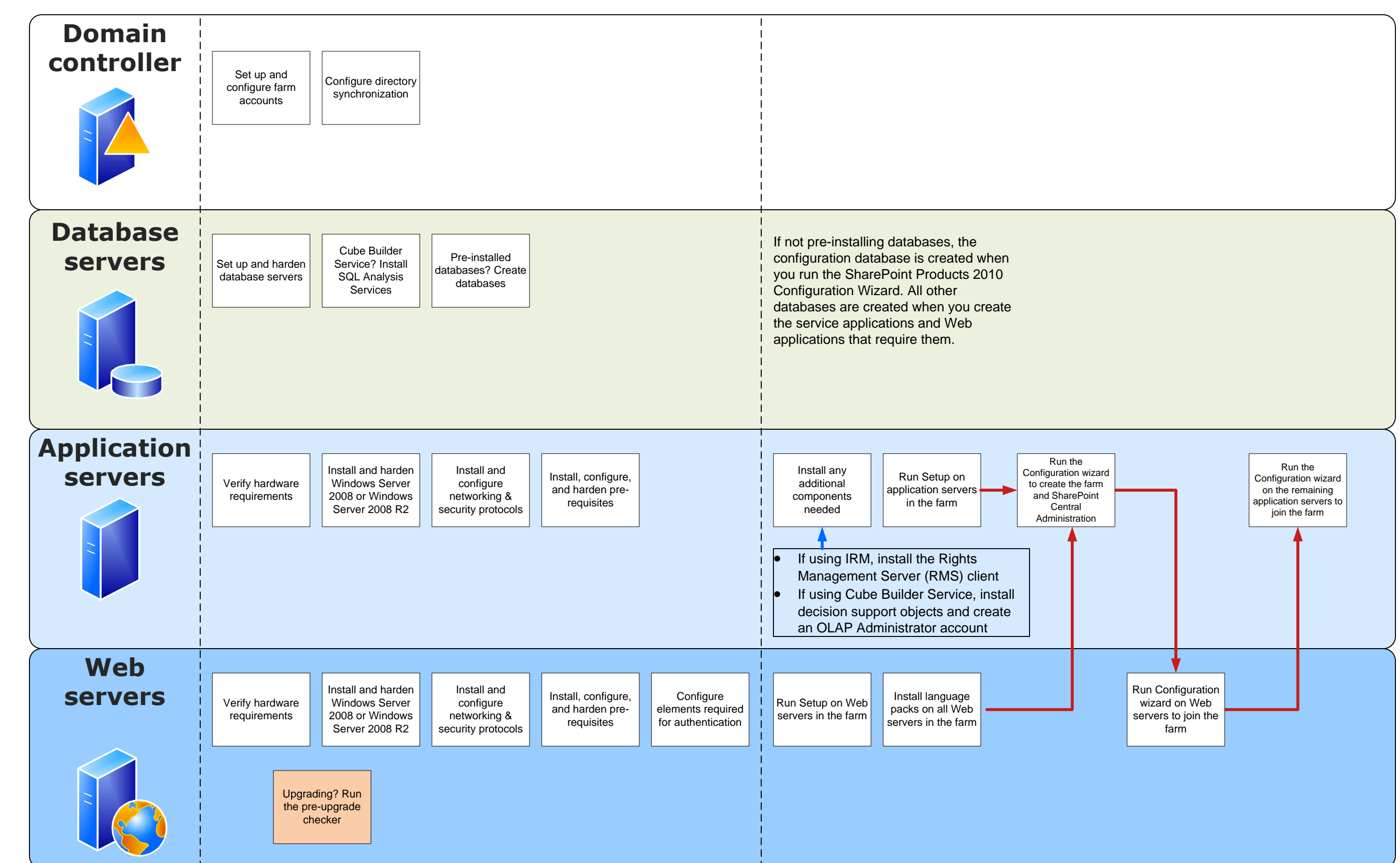
There are three phases in the process of installing and configuring your environment:

### 1 Prepare the servers

In this phase, you get your servers ready to host the product.

### 2 Install and build the farm

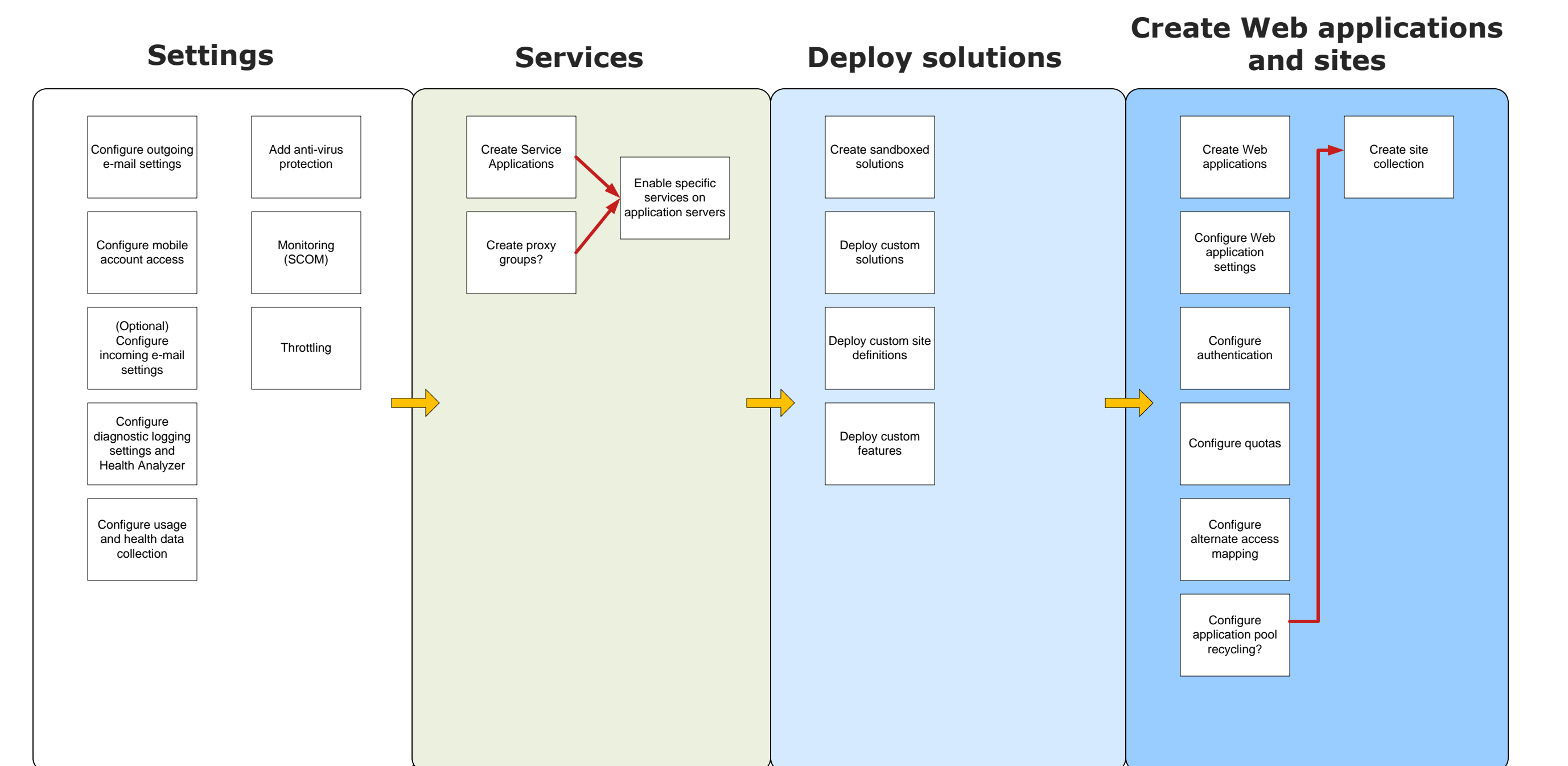
In this phase, you install the product and assign roles to each server. You also create the configuration database and the SharePoint Central Administration Web site.



### 3 Configure settings, services, solutions, and sites

In this phase, you prepare the farm to host your site content by configuring global settings, creating services applications, deploying customizations, and creating and populating the sites.

You can use the Farm Configuration Wizard to these configuration steps, or you can perform them by using either the SharePoint Central Administration Web site or Windows PowerShell. Configuration steps are not isolated to a specific tier in the server infrastructure.



These steps include only a high-level overview of the process. To successfully plan and perform your deployment, follow the recommendations and procedures in the [Planning Guide](#) and the [Deployment Guide](#) for SharePoint Server 2010 or SharePoint Foundation 2010.