

Install MongoDB Community Edition on Windows¶

On this page

Overview

Requirements

Get MongoDB Community Edition

Install MongoDB Community Edition

Run MongoDB Community Edition

Configure a Windows Service for MongoDB Community Edition

Manually Create a Windows Service for MongoDB Community Edition

Additional Resources

Overview¶

Use this tutorial to install MongoDB Community Edition on Windows systems.

Platform Support

Starting in version 3.6 MongoDB requires Windows Server 2008 R2, Windows 7, or later.

Important

If you are running any edition of Windows Server 2008 R2 or Windows 7, please install a hotfix to resolve an issue with memory mapped files on Windows.

Note

Starting in MongoDB 3.6, MongoDB binaries, mongod and mongos, bind to localhost by default. Previously, starting in MongoDB 2.6, only the binaries from the official MongoDB RPM (Red Hat, CentOS, Fedora Linux, and derivatives) and DEB (Debian, Ubuntu, and derivatives) packages bind to localhost by default. For more details, see [Localhost Binding Compatibility Changes](#).

Requirements¶

MongoDB Community Edition requires Windows Server 2008 R2, Windows 7, or later. The .msi installer includes all other software dependencies and will automatically upgrade any older version of MongoDB installed using an .msi file.

To find which version of Windows you are running, enter the following commands in the Command Prompt or Powershell:

```
wmic os get caption
```

```
wmic os get osarchitecture
```

Get MongoDB Community Edition¶

Note

To install a different version of MongoDB, please refer to that version's documentation. For example, see [version 3.4](#).

Download the latest production release of MongoDB from the MongoDB Download Center.

Install MongoDB Community Edition¶

Interactive Installation¶

In Windows Explorer, locate the downloaded MongoDB .msi file, which typically is located in the default Downloads folder. Double-click the .msi file. A set of screens will appear to guide you through the installation process.

You may specify an installation directory if you choose the “Custom” installation option.

Note

These instructions assume that you have installed MongoDB to C:\Program Files\MongoDB\Server\3.6\.

During the installation process you will be given the option to install MongoDB Compass in addition to MongoDB Server.

MongoDB is self-contained and does not have any other system dependencies. You can run MongoDB from any folder you choose. You may install MongoDB in any folder (e.g. D:\test\mongodb).

Unattended Installation¶

You may install MongoDB Community unattended on Windows from the command line using msixec.exe.

1

Open an Administrator command prompt.¶

Press the Win key, type cmd.exe, and press Ctrl + Shift + Enter to run the Command Prompt as Administrator.

Execute the remaining steps from the Administrator command prompt.

2

Install MongoDB for Windows.¶

Change to the directory containing the .msi installation binary of your choice and invoke:

Copy

```
msiexec.exe /q /i mongodb-win32-x86_64-2008plus-ssl-3.6.2-signed.msi ^
```

```
    INSTALLLOCATION="C:\Program Files\MongoDB\Server\3.6.2\" ^
```

```
    ADDLOCAL="all"
```

You can specify the installation location for the executable by modifying the INSTALLLOCATION value.

By default, this method installs all MongoDB binaries. To install specific MongoDB component sets, you can specify them in the ADDLOCAL argument using a comma-separated list including one or more of the following component sets:

Component Set	Binaries
Server	mongod.exe
Router	mongos.exe
Client	mongo.exe
MonitoringTools	mongostat.exe, mongotop.exe
ImportExportTools	mongodump.exe, mongorestore.exe, mongoexport.exe, mongoimport.exe
MiscellaneousTools	bsondump.exe, mongofiles.exe, mongoperf.exe

For instance, to install only the MongoDB utilities, invoke:

Copy

```
msiexec.exe /q /i mongodb-win32-x86_64-2008plus-ssl-3.6.2-signed.msi ^
```

```
    INSTALLLOCATION="C:\Program Files\MongoDB\Server\3.6.2\" ^
```

```
    ADDLOCAL="MonitoringTools,ImportExportTools,MiscellaneousTools"
```

Run MongoDB Community Edition¶

Warning

Do not make exe visible on public networks without running in “Secure Mode” with the auth setting. MongoDB is designed to be run in trusted environments, and the database does not enable “Secure Mode” by default.

1

Set up the MongoDB environment.¶

MongoDB requires a data directory to store all data. MongoDB’s default data directory path is the absolute path \data\db on the drive from which you start MongoDB. Create this folder by running the following command in a Command Prompt:

Copy

```
md \data\db
```

You can specify an alternate path for data files using the --dbpath option to exe, for example:

Copy

```
"C:\Program Files\MongoDB\Server\3.6\bin\mongod.exe" --dbpath d:\test\mongodb\data
```

If your path includes spaces, enclose the entire path in double quotes, for example:

Copy

```
"C:\Program Files\MongoDB\Server\3.6\bin\mongod.exe" --dbpath "d:\test\mongo db data"
```

You may also specify the dbpath in a configuration file.

2

Start MongoDB.¶

To start MongoDB, run exe. For example, from the Command Prompt:

Copy

```
"C:\Program Files\MongoDB\Server\3.6\bin\mongod.exe"
```

This starts the main MongoDB database process. The waiting for connections message in the console output indicates that the exe process is running successfully.

Depending on the security level of your system, Windows may pop up a Security Alert dialog box about blocking “some features” of C:\Program Files\MongoDB\Server\3.6\bin\mongod.exe from communicating on networks. All users should select Private Networks, such as my home or work network and click Allow access. For additional information on security and MongoDB, please see the Security Documentation.

3

Verify that MongoDB has started successfully¶

Verify that MongoDB has started successfully by checking the process output for the following line:

```
[initandlisten] waiting for connections on port 27017
```

The output should be visible in the terminal or shell window.

You may see non-critical warnings in the process output. As long as you see the log line shown above, you can safely ignore these warnings during your initial evaluation of MongoDB.

4

Connect to MongoDB.¶

To connect to MongoDB through the ~bin.mongo.exe shell, open another Command Prompt.

Copy

```
"C:\Program Files\MongoDB\Server\3.6\bin\mongo.exe"
```

If you want to develop applications using .NET, see the documentation of C# and MongoDB for more information.

Configure a Windows Service for MongoDB Community Edition¶

1

Open an Administrator command prompt.¶

Press the Win key, type cmd.exe, and press Ctrl + Shift + Enter to run the Command Prompt as Administrator.

Execute the remaining steps from the Administrator command prompt.

2

Create directories.¶

Create directories for your database and log files:

Copy

```
mkdir c:\data\db
```

```
mkdir c:\data\log
```

3

Create a configuration file.¶

Create a configuration file. The file must set `systemLog.path`. Include additional configuration options as appropriate.

For example, create a file at `C:\Program Files\MongoDB\Server\3.6\mongod.cfg` that specifies both `systemLog.path` and `storage.dbPath`:

Copy

```
systemLog:
```

```
    destination: file
```

```
    path: c:\data\log\mongod.log
```

```
storage:
```

```
    dbPath: c:\data\db
```

4

Install the MongoDB service.¶

Important

Run all of the following commands in Command Prompt with “Administrative Privileges”.

Install the MongoDB service by starting `exe` with the `--install` option and the `-config` option to specify the previously created configuration file.

Copy

```
"C:\Program Files\MongoDB\Server\3.6\bin\mongod.exe" --config "C:\Program Files\MongoDB\Server\3.6\mongod.cfg" --install
```

To use an alternate `dbpath`, specify the path in the configuration file (e.g. `C:\mongodb\mongod.cfg`) or on the command line with the `--dbpath` option.

If needed, you can install services for multiple instances of `exe` or `exe`. Install each service with a unique `--serviceName` and `--serviceDisplayName`. Use multiple instances only when sufficient system resources exist and your system design requires it.

5

Start the MongoDB service.¶

Copy

```
net start MongoDB
```

6

Verify that MongoDB has started successfully¶

Verify that MongoDB has started successfully by checking the log file at `c:\data\log\mongod.log` for the following line:

`[initandlisten] waiting for connections on port 27017`

If you modified the `systemLog.path` configuration file option, open the log file at that location to examine the process output.

You may see non-critical warnings in the process output. As long as you see the log line shown above, you can safely ignore these warnings during your initial evaluation of MongoDB.

7

Connect to MongoDB.¶

To connect to MongoDB through the `~bin.mongo.exe` shell, open another Command Prompt.

Copy

`"C:\Program Files\MongoDB\Server\3.6\bin\mongo.exe"`

If you want to develop applications using .NET, see the documentation of C# and MongoDB for more information.

8

Stop or remove the MongoDB service as needed.¶

To stop the MongoDB service use the following command:

Copy

`net stop MongoDB`

To remove the MongoDB service use the following command:

Copy

`"C:\Program Files\MongoDB\Server\3.6\bin\mongod.exe" --remove`

Manually Create a Windows Service for MongoDB Community Edition¶

You can set up the MongoDB server as a Windows Service that starts automatically at boot time.

The following procedure assumes you have installed MongoDB Community using the .msi installer with the path `C:\Program Files\MongoDB\Server\3.6\`.

If you have installed in an alternative directory, you will need to adjust the paths as appropriate.

1

Open an Administrator command prompt.¶

Press the Win key, type `cmd.exe`, and press `Ctrl + Shift + Enter` to run the Command Prompt as Administrator.

Execute the remaining steps from the Administrator command prompt.

2

Create directories.¶

Create directories for your database and log files:

Copy

```
mkdir c:\data\db
```

```
mkdir c:\data\log
```

3

Create a configuration file.¶

Create a configuration file. The file must set `systemLog.path`. Include additional configuration options as appropriate.

For example, create a file at `C:\Program Files\MongoDB\Server\3.6\mongod.cfg` that specifies both `systemLog.path` and `storage.dbPath`:

Copy

`systemLog:`

`destination: file`

`path: c:\data\log\mongod.log`

`storage:`

`dbPath: c:\data\db`

4

Create the MongoDB service.¶

Create the MongoDB service.

Copy

```
sc.exe create MongoDB binPath= "\"C:\Program Files\MongoDB\Server\3.6\bin\mongod.exe\"" --  
service --config= "\"C:\Program Files\MongoDB\Server\3.6\mongod.cfg\"" DisplayName= "MongoDB"  
start= "auto"
```

`sc.exe` requires a space between `"="` and the configuration values (eg `"binPath= "`), and a `"` to escape double quotes.

If successfully created, the following log message will display:

```
[SC] CreateService SUCCESS
```

5

Start the MongoDB service.¶

Copy

net start MongoDB

6

Verify that MongoDB has started successfully¶

Verify that MongoDB has started successfully by checking the log file at c:\data\log\mongod.log for the following line:

[initandlisten] waiting for connections on port 27017

If you modified the systemLog.path configuration file option, open the log file at that location to examine the process output.

You may see non-critical warnings in the process output. As long as you see the log line shown above, you can safely ignore these warnings during your initial evaluation of MongoDB.

7

Connect to MongoDB.¶

To connect to MongoDB through the ~bin.mongo.exe shell, open another Command Prompt.

Copy

"C:\Program Files\MongoDB\Server\3.6\bin\mongo.exe"

If you want to develop applications using .NET, see the documentation of C# and MongoDB for more information.

8

Stop or remove the MongoDB service as needed.¶

To stop the MongoDB service, use the following command:

net stop MongoDB

To remove the MongoDB service, first stop the service and then run the following command:

sc.exe delete MongoDB