



We use optional cookies to improve your experience on our websites, such as through social media connections, and to display personalized advertising based on your online activity. If you reject optional cookies, only cookies necessary to provide you the services will be used. You may change your selection by clicking "Manage Cookies" at the bottom of the page. [Privacy Statement](#) [Third-Party Cookies](#)

Accept

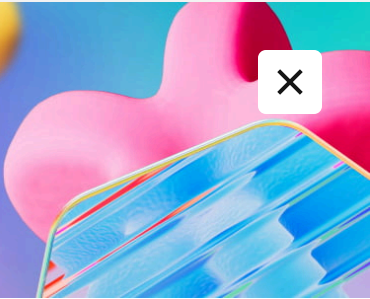
Reject

Manage cookies

# Microsoft Ignite

Nov 19–22, 2024

Register now >



## Add-BitsFile

Reference

Feedback

Module: [BitsTransfer](#)

### In this article

- [Syntax](#)
- [Description](#)
- [Examples](#)
- [Parameters](#)
- [Show 3 more](#)

Adds one or more files to an existing BITS transfer job.

## Syntax

```
Add-BitsFile  
  [-BitsJob] <BitsJob[]>  
  [[-Destination] <String[]>]  
  [-Source] <String[]>  
  [-WhatIf]  
  [-Confirm]  
  [<CommonParameters>]
```

## Description

The **Add-BitsFile** cmdlet adds files to a Background Intelligent Transfer Service (BITS) transfer job. You can specify the files to add to the BITS transfer job by name at the command prompt or in a comma-separated value (CSV) file.

Important: An upload job can contain only one file. To upload more than one file, use the **Import-Csv** cmdlet, and pipe the output to the **Add-BitsFile** cmdlet. For more information, see example 3 in this Help topic. Or, use a cabinet file (.cab) or a compressed file (.zip).

## Examples

### Example 1: Append a file to the transfer queue of an existing BITS transfer job

```
PS C:\> Get-BitsTransfer -JobId 10778CFA-C1D7-4A82-8A9D-80B1
```

This command appends a file to the transfer queue of an existing BITS transfer job.

In this example, the output of the **Get-BitsTransfer** cmdlet is a **BitsJob** object that is identified by its unique job ID. The command pipes the job ID to the **Add-BitsFile** cmdlet. The local and remote names of the file are in the parameters.

## Example 2: Append a set of files to the transfer queue of an existing BITS transfer job

```
PS C:\> $Bits = Get-BitsTransfer -JobId 10778CFA-C1D7-4A82-8
PS C:\> Add-BitsFile -BitsJob $Bits -Source "http://server01
```

This command appends a set of files to the transfer queue of an existing BITS transfer job.

The first command retrieves the BITS transfer job that is identified by the job ID and then stores it in the `$b` variable. The second command uses the *BitsJob* parameter to pass the **BitsJob** object that is stored in the `$b` variable to **Add-BitsFile**.

The server file names are paired with the corresponding client file names.

## Example 3: Add a set of files to the transfer queue of a new BITS transfer job

```
PS C:\> $Bits = Start-BitsTransfer -Suspended
PS C:\> Import-CSV filelist.txt | Add-BitsFile -BitsJob $Bit
PS C:\> Resume-BitsTransfer -BitsJob $Bits
```

This example adds a set of files to the transfer queue of a new BITS transfer job.

The first command creates a new **BitsJob** object and then stores it in the **\$Bits** variable.

The second command uses the **Import-CSV** cmdlet to import a text file that contains a list of files to be transferred. The text file is converted to an array of objects (one per line) and passed through the pipeline to the **Add-BitsFile** cmdlet. The *BitsJob* parameter is used to pass the **BitsJob** object (the transfer job) that is stored in the **\$Bits** variable to the **Add-BitsFile** cmdlet. These array of objects, adds BITS transfer job for each of the file to be downloaded and then transfers them concurrently to the client. This command also updates the transfer job with the list of files to be transferred.

The third command passes the **BitsJob** object that is stored in the **\$Bits** variable to the **Resume-BitsTransfer** cmdlet. The BITS transfer job is restarted, and the files that are specified in the Filelist.txt file are transferred from the source to the destination.

The `Import-CSV filelist.txt` element of the second command imports a text file that contains the list of files to be transferred. Each line of this file specifies a file to be transferred, in the `<Source>`, `<Destination>` format. The text file is converted to an array of objects, one per line, and passed through the pipeline. In this example, the array of objects is passed to the `Add-BitsFile` cmdlet.

The contents of the Filelist.txt file resemble the following information:

- **Source, Destination**
- `http://server01/servertestdir/testfile1.txt,`  
`c:\clienttestdir\testfile1.txt`
- `http://server01/servertestdir/testfile2.txt,`  
`c:\clienttestdir\testfile2.txt`
- `http://server01/servertestdir/testfile3.txt,`  
`c:\clienttestdir\testfile3.txt`

- `http://server01/servertestdir/testfile4.txt,`  
`c:\clienttestdir\testfile4.txt`

# Parameters

## -BitsJob

Specifies the BITS transfer jobs to which you want to add files. You can pipe a value to this parameter from other cmdlets that return **BitsJob** objects, such as **Get-BitsTransfer**.

 Expand table

Type:	BitsJob[]
Position:	0
Default value:	None
Required:	True
Accept pipeline input:	True
Accept wildcard characters:	False

## -Confirm

Prompts you for confirmation before running the cmdlet.

 Expand table

Type:	SwitchParameter
Aliases:	cf
Position:	Named
Default value:	False
Required:	False

Accept pipeline input:	False
Accept wildcard characters:	False

**-Destination**

Specifies the destination location and the names of the files that you want to transfer. The destination names are paired with the corresponding source file names. For instance, the first file name specified in the *Source* parameter corresponds to the first file name in the *Destination* parameter, and the second file name in the *Source* parameter corresponds to the second file name in the *Destination* parameter. The *Source* and *Destination* parameters must have the same number of elements; otherwise, the command produces an error.

 Expand table

Type:	String[]
Position:	2
Default value:	None
Required:	False
Accept pipeline input:	True
Accept wildcard characters:	False

**-Source**

Specifies the source location and the names of the files that you want to transfer. The source file names are paired with the corresponding destination file names. For instance, the first file name specified in the *Source* parameter corresponds to the first file name in the *Destination* parameter, and the second file name in the *Source* parameter corresponds to the second file name in the

*Destination* parameter. The *Source* and *Destination* parameters must have the same number of elements; otherwise, the command produces an error. You can use standard wildcard characters such as the asterisk (\*) and the question mark (?), or you can use a range operator such as "[a-r]".

 Expand table

Type:	String[]
Position:	1
Default value:	None
Required:	True
Accept pipeline input:	True
Accept wildcard characters:	False

**-WhatIf**

Shows what would happen if the cmdlet runs. The cmdlet is not run.

 Expand table

Type:	SwitchParameter
Aliases:	wi
Position:	Named
Default value:	False
Required:	False
Accept pipeline input:	False
Accept wildcard characters:	False

## Inputs

Microsoft.BackgroundIntelligentTransfer.Management.BitsJob[]

This cmdlet accepts one or more **BitsJob** objects as input that populates the *BitsJob* parameter.

## Outputs

Microsoft.BackgroundIntelligentTransfer.Management.BitsJob[]

This cmdlet generates the **BitsJob** objects that are associated with the BITS transfer jobs to which the files were added.

## Related Links

- [Complete-BitsTransfer](#)
- [Get-BitsTransfer](#)
- [Remove-BitsTransfer](#)
- [Resume-BitsTransfer](#)
- [Set-BitsTransfer](#)
- [Start-BitsTransfer](#)
- [Suspend-BitsTransfer](#)

## Feedback




Was this page helpful?

 Yes

 No

[Provide product feedback](#) 



[Manage cookies](#)   [Previous Versions](#)   [Blog](#)    [Contribute](#)   [Privacy](#)    [Terms of Use](#)   [Trademarks](#) 

© Microsoft 2024