

# UpgradeSolution CLI

Upgrade FrameworkX solution files from the command line.

[Support Resources](#) [Legacy Versions](#) [Upgrade](#) UpgradeSolution CLI

Version 10.1.5+

UpgradeSolution.exe performs the same operation as the **Upgrade** button in the Solution Center, without launching the UI. Useful for scripted upgrades, batch migration of many solutions, and unattended deployment pipelines.

[Usage](#)  
[Behavior](#)  
[Exit Codes](#)  
[Installation](#)  
[Examples](#)  
[See also](#)

## Usage

```
UpgradeSolution <solution>
```

Running with no arguments prints usage help and exits with code 2.

## Argument forms

Form	Example	Resolution
Full path to a .dbsln file	UpgradeSolution "C:\Projects\MyPlant.dbsln"	Used as-is.
Solution name only	UpgradeSolution MyPlant	Resolved under My Documents\FrameworkX\Solutions\MyPlant.dbsln.
Solution name with extension	UpgradeSolution MyPlant.dbsln	Same as above.

The .dbsln extension is appended automatically if omitted.

## Behavior

- Reads the current ProductVersion from the solution's InfoProjectVersion table.
- Calls InfoServer.BeginUpgradeProject, polls PartialUpgradeProject, then calls EndUpgradeProject. No UI, no remoting stack.
- Writes progress lines to standard output in the form [ 42%] Upgrading table 'X'....
- A backup of the original solution is created automatically by BeginUpgradeProject. The backup file sits next to the original with the same name pattern the UI button produces.

## Exit Codes

Code	Meaning
0	Success. Solution upgraded.
1	Upgrade error. See standard error for detail.
2	Bad arguments. Missing solution argument, file not found, or malformed input.

## Installation

UpgradeSolution.exe ships with the 10.1.5 installer inside the FactoryStudio folder, alongside the other CLI utilities.

On first launch of the FrameworkX Solution Center after install, a launcher script `UpgradeSolution.bat` is auto-deployed to `My Documents\FrameworkX\Utilities\`. The launcher mirrors the pattern used by `StartWebServices.bat`, `SolutionCreator.bat`, and `RemoveAllServices.bat`.

Call either the executable directly or the launcher script. The launcher forwards all arguments to the executable.

---

## Examples

### Upgrade a single solution by full path

```
& "C:\Program Files\Tatsoft\FrameworkX\fx-10\UpgradeSolution.exe" `
"C:\Projects\MyPlant.dbsln"
```

### Upgrade by solution name

```
UpgradeSolution MyPlant
```

Resolves under `My Documents\FrameworkX\Solutions\MyPlant.dbsln`.

### Batch upgrade a folder of solutions

```
Get-ChildItem "D:\Plants\*.dbsln" | ForEach-Object {
    Write-Host "Upgrading ${_}.Name)"
    & "C:\Program Files\Tatsoft\FrameworkX\fx-10\UpgradeSolution.exe" $_.FullName
    if ($LASTEXITCODE -ne 0) {
        Write-Error "Failed: ${_}.Name), exit $LASTEXITCODE"
    }
}
```

---

## See also

- [Legacy Versions Upgrade](#), parent page for upgrade procedures.
- [Upgrading Solutions Reference](#), UI-driven upgrade through the Solution Center.

---

## In this section...

---