

React Native for Windows + macOS 0.72

Docs

**APIs** 

Blog

Resources

Samples

Support

**NATIVE DEVELOPMENT (WINDOWS)** 

# Managing C++ dependencies

**Edit** 

Details to consider when consuming community modules or other Visual C++ projects.

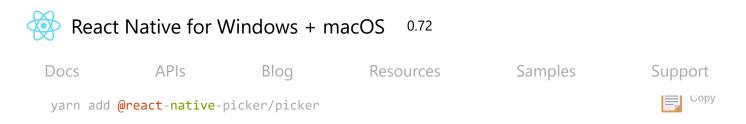
#### **Details**

Applications with native code, either written on C# or C++, may add source dependencies on native Visual C++ ( .vcxproj ) projects. Meaning, such dependencies will be built as part of the application.

Starting with version 0.68, React Native for Windows apps use the <a href="PackageReference">PackageReference</a> restore project style for native C++ NuGet dependencies. The main change consists of NuGet packages being placed directly in the user account's <a href="globalPackagesFolder">globalPackagesFolder</a> cache instead of also copying them into a local folder relative to the Visual Studio Solution location (see <a href="repositoryPath">repositoryPath</a>).

This may conflict with C++ dependencies that use the more common <a href="mailto:packages.config">packages.config</a> project style, including community modules generated targeting versions older than 0.68.

## **Updating your app**



### Add a packages.config file to your app's project directory

The React Native Windows build system will try to build the dependency from source.

Because it uses the packages.config restore style, it will most likely expect its own NuGet dependencies to be restored at \$(SolutionDir)packages\.

This way, your React Native Windows app will restore the external NuGet dependencies into the expected location.

## **Building with MSBuild.exe (Advanced)**

The introduction of PackageReference in React Native Windows projects overrides the packages.config NuGet restore style in the MSBuild engine.



#### React Native for Windows + macOS

Docs

**APIs** 

Blog

Resources

Samples

Support

To successfully restore and build your app using only the MSBuild CLI, run the following commands:

MSBuild.exe /t:Restore "/p:RestoreProjectStyle=PackagesConfig;RestorePackagesConfi



MSBuild.exe /restore your solution.sln



The first command explicitly restores any packages.config dependencies found within your solution at the predefined destination (usually, the packages folder in your solution's path).

The second command implicitly restores and builds the solution using the now default restore style, PackageReference, restoring the remaining NuGet dependencies.

Note, these steps are done automatically when building inside Visual Studio or using the React Native Windows CLI.

< Customizing SDK versions

Get Started with macOS >

**REACT NATIVE DOCS** 

**Getting Started** 

Tutorial

Components and APIs

More Resources

**REACT NATIVE FOR WINDOWS +** 

**MACOS DOCS** 

**Get Started with Windows** 

Get Started with macOS

**React Native Windows Components** 

and APIs

Native Modules

Native UI Components

**CONNECT WITH US ON** 

Blog

Twitter

GitHub

Samples