

# 第77期-使用文件变化监控程序开发应用

2019年12月17日 20:05

## 文件观察程序介绍

微软提供一个 dotnet watch 命令用于监控项目文件的变更，当文件变更时可以触发事先配置好的命令的工具，在持续集成领域可以广泛应用。

例如：您希望文件更改时自动触发编译命令、并自动化执行测试，然后自动化部署。

## 使用 dotnet watch 运行 .NET Core CLI 命令

| 命令                                    | 带 watch 的命令                                 |
|---------------------------------------|---|
| dotnet run                            | dotnet watch run                            |
| dotnet run -f netcoreapp2.0           | dotnet watch run -f netcoreapp2.0           |
| dotnet run -f netcoreapp2.0 -- --arg1 | dotnet watch run -f netcoreapp2.0 -- --arg1 |
| dotnet test                           | dotnet watch test                           |

`dotnet watch --project <PROJECT>` 来指定要监视的项目

## 自定义要监视的文件列表

技术点：glob 模式文件匹配模式

默认情况下，dotnet-watch 跟踪与以下 glob 模式匹配的所有文件：

- `**/*.cs`
- `*.csproj`
- `**/*.resx`

```
<ItemGroup>
  <!-- extends watching group to include *.js files -->
  <Watch Include="**/*.js" Exclude="node_modules\**\*;**/*.js.map;obj\**\*;bin\**\*" />
</ItemGroup>
```

## 从要监视的文件中排除

```

<ItemGroup>
    <!-- exclude Generated.cs from dotnet-watch -->
    <Compile Include="Generated.cs" Watch="false" />

    <!-- exclude Strings.resx from dotnet-watch -->
    <EmbeddedResource Include="Strings.resx" Watch="false" />

    <!-- exclude changes in this referenced project -->
    <ProjectReference Include="..\ClassLibrary1\ClassLibrary1.csproj"
Watch="false" />
</ItemGroup>

```

## 自定义监视项目

如果目标是同时监视两个项目，可创建一个自定义项目文件。

```

<Project>
    <ItemGroup>
        <TestProjects Include="**\*.csproj" />
        <Watch Include="**\*.cs" />
    </ItemGroup>

    <Target Name="Test">
        <MSBuild Targets="VSTest" Projects="@ (TestProjects)" />
    </Target>

    <Import Project="$(MSBuildExtensionsPath)\Microsoft.Common.targets" />
</Project>

```

**dotnet watch msbuild /t:Test**

## 深入阅读

如果你对 dotnet-watch 工具的工作原理感兴趣，其实实现部分很简单 dotnet-watch 源码中使用 DotnetFileWatcher 和 PollingFileWatcher 类来实现对文件系统中文件的变更检测，其中 DotnetFileWatcher 适用于有 FileSystemWatcher 功能实现的操作系统（比如：windows 系列操作系统），而 PollingFileWatcher 适用于没有内建 FileSystemWatcher 功能的操作系统（比如：Linux），通过轮询的方式定时检查目录中文件的最后写入时间来判断文件是否发生了变更，一旦发现有文件发生改变，该工具将重新编译、启动应用。

<https://github.com/aspnet/AspNetCore/tree/master/src/Tools/dotnet-watch>

