

A quick note about WRL's ChainInterfaces template class

 devblogs.microsoft.com/oldnewthing/20230501-00

May 1, 2023



Raymond Chen

The Windows Runtime C++ Template Library (commonly known as WRL) contains a template class called `ChainInterfaces`. The documentation for `ChainInterfaces` talks about what it does but doesn't tell you *why it's there* or when you should use it.

The purpose of `ChainInterfaces` is to be included among the template arguments to the WRL `RuntimeClass` and `Implements` template classes to indicate that you have a sequence ("chain") of interfaces where each one extends the previous one.

For example, the `IFileSystemBindData2` interface extends the `IFileSystemBindData` interface. If you want to use WRL to implement an object that implements both interfaces, you would write

```
namespace wrl = Microsoft::WRL;

struct MyFileSystemBindData :
    wrl::RuntimeClass<
        wrl::RuntimeClassFlags<wrl::ClassicCom>,
        wrl::ChainInterfaces<IFileSystemBindData2, IFileSystemBindData>
    >
{
    [ implementation elided for expository purposes ]
}
```

Note that you list the `ChainInterfaces` template parameters from *most derived to least derived*. Fortunately, if you get the order wrong, you get a compile-time error.