

Enumerating Windows clipboard history in C++/WinRT and C#

 devblogs.microsoft.com/oldnewthing/20230302-00

March 2, 2023



Raymond Chen

Here's a console program that dumps all the text from the Windows clipboard history:

```
#include <stdio.h>
#include <winrt/Windows.ApplicationModel.DataTransfer.h>
#include <winrt/Windows.Foundation.h>
#include <winrt/Windows.Foundation.Collections.h>

namespace winrt
{
    using namespace winrt::Windows::Foundation;
    using namespace winrt::Windows::Foundation::Collections;
    using namespace winrt::Windows::ApplicationModel::DataTransfer;
}

winrt::IAsyncAction DumpClipboardHistoryAsync()
{
    auto result = co_await Clipboard::GetHistoryItemsAsync();
    auto items = result.Items();
    for (auto item : items) {
        auto content = item.Content();
        if (content.Contains(StandardDataFormats::Text())) {
            auto text = co_await content.GetTextAsync();
            printf("%ls: %ls\n", item.Id().c_str(), text.c_str());
        } else {
            printf("%ls: (no text content)\n", item.Id().c_str());
        }
    }
}

int main()
{
    winrt::init_apartment(winrt::apartment_type::multi_threaded);

    DumpClipboardHistoryAsync().get();

    return 0;
}
```

First, we ask for all of the clipboard history items. This call may fail and return no items; if we are interested in that case, we can check the `ClipboardHistoryItemResultStatus` to see why we got nothing. (Maybe the user hasn't enabled clipboard history.)

Conveniently, if clipboard history is not available, the `GetHistoryItemsAsync` method produces an empty vector, so we can just go straight to the vector of results.¹

For each item in the results, we ask if its content contains text. If so, then we ask for the text and print it. (If not, then we just say "Sorry.")

The sample doesn't demonstrate it, but you can also take a history item and pass it to `Clipboard::SetHistoryItemAsContent` to make it the current item on the clipboard.

And here's the C# version:

```
using System;
using System.Threading.Tasks;
using Windows.ApplicationModel.DataTransfer;

class Program
{
    public static async Task Main()
    {
        var result = await Clipboard.GetHistoryItemsAsync();
        var items = result.Items;
        foreach (var item in items) {
            var content = item.Content;
            if (content.Contains(StandardDataFormats.Text)) {
                var text = await content.GetTextAsync();
                System.Console.WriteLine($"{item.Id}: {text}");
            } else {
                System.Console.WriteLine($"{item.Id}: (no text content)");
            }
        }
    }
}
```

It's a straightforward translation of the C++/WinRT version.

Next time, we'll translate this into PowerShell.

Wait, what?

Yup, PowerShell.

¹ This is a general principle in the Windows Runtime that methods which produce vectors return empty vectors rather than null pointers on failure.

