



PowerShell Conference Europe

Next Up:

Profiler internals

Jakub Jareš

3

2

1



PowerShell Conference Europe

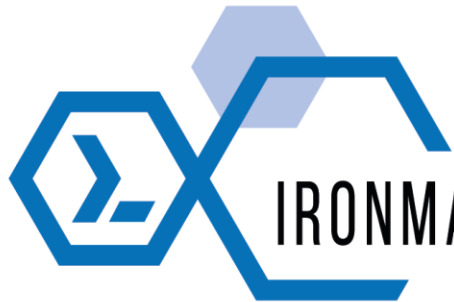
Profiler internals

Jakub Jareš

Many thanks to our sponsors:



Jane Street



IRONMAN SOFTWARE

KNOWLEDGE
FACTORY



SynEdgy



@jakubjares.com



Intermission

<https://bepug.odoo.com/speed-up-the-code>

Attending this session will not help, unless you want to hack PowerShell (which they did not ban, yet! 🤪).

But try this:

https://www.youtube.com/playlist?list=PL_47I_1HjEMJfMMsYBLIY35T7eLOyJz3C



@jakubjares.com



PSCONF EU SPEAKER

@jakubjares.com

@nohwnd

@pspester

me@jakubjares.com

Jakub Jareš



Pester and Profiler owner and maintainer.

Senior software engineer, developing VSTest, Testing Platform and MSTest at Microsoft. All opinions are mine.

Please consider sponsoring my open-source development:

[Sponsor @nohwnd on GitHub Sponsors](#)



@jakubjares.com

Profiler

Finds slow PowerShell
code, automagically!

You shall not be, **not fast!**



Speed the fudge up!



User experience!

SelfPercent SelfDuration

HitCount File

Line Text

91.84 00:00:00.1648102

2 Avocado.psm1

69 \$r = Invoke-RestMethod -Method GET -Uri \$uri -UseBasicAuthentication

4.88 00:00:00.0007824

1 Write-Avocado.ps1

1 Import-Module \$PSScriptRoot\Avocado.psm1 -Force

2.73 00:00:00.0048939

1 01_trace avocado.ps1

2 & "\$PSScriptRoot/Write-Avocado.ps1"

0.19 00:00:00.0003396

2 Avocado.psm1

70 \$r | Where-Object { \$_.annotation -eq \$Emoji } | Select-Object

0.13 00:00:00.0002393

1 Avocado.psm1

19 Export-ModuleMember -Function Get-Avocado, Get-Unicode



How to make a profiler?

Fire **every time** something happens in the script know:

- **what** happened
- **when** it happened
- find what happened most often / took most time

How to make a profiler?

Sampling



How to make a profiler?

Events



How to do that in PowerShell?

Finding a way to hook trigger

- <https://grep.app/search?f.repo=PowerShell%2FPowerShell&f.repo.pattern=powershell&q=%3E%3E%3E%3E>
- <https://github.com/PowerShell/PowerShell/blob/master/src/System.Management.Automation/engine/debugger/debugger.cs#L4153> - TraceLine
- <https://github.com/PowerShell/PowerShell/pull/13589>
- <https://github.com/pardeike/Harmony>

Patching UI

- <https://github.com/nohwnd/Profiler/blob/main/csharp/Profiler/TracerHostUI.cs>
- <https://github.com/nohwnd/Profiler/blob/827281b1a6b31ddf37c203dcd0171947e382868f/Profiler/TraceScriptInternal.ps1#L94>
- <https://github.com/nohwnd/Profiler/blob/main/csharp/Profiler/Tracer.cs#L127>

Trigger is hooked

```
C:\> "hello"  
DEBUG: 1+ >>>> "hello"  
hello  
C:\>
```

\$host.UI -> internalUI -> externalUI

\$host.UI -> internalUI -> **profilerUI** -> externalUI



```
$events += @{  
    what = 'DEBUG: 1+ >>>> "hello"'  
    when = [Diagnostics.Stopwatch]::GetTimestamp()  
}
```

Getting more info

- What exactly are we calling?
- Where in the scriptblock are we?
- Which function is being called?
- From which module?
- How deep are we in the call stack?
- Are we recursing?

How to jump up?

```
hell.ConsoleHostUserInterface.WriteLine(string message) Line 721
nt.Automation.Internal.Host.InternalHostUserInterface.WriteLineHelper(string message) Line 215
nt.Automation.Internal.Host.InternalHostUserInterface.WriteLine(string message, ref System.Management.Automation.ActionPreference pre
nt.Automation.ScriptDebugger.TraceLine(System.Management.Automation.Language.IScriptExtent extent) Line 2476
nt.Automation.ScriptDebugger.OnSequencePointHit(System.Management.Automation.Language.FunctionContext functionContext) Line 855
nt.Automation.ScriptDebugger.EnterScriptFunction(System.Management.Automation.Language.FunctionContext functionContext) Line 577
nt.Automation.Interpreter.ActionCallInstruction<System.Management.Automation.ScriptDebugger, System.Management.Automation.Language.Fu
```

\$ExecutionContext	._context	.Debugger	._callStack	.LastFunctionContext()
EngineIntrinsics	ExecutionContext	ScriptDebugger	CallStackList	FunctionContext

<https://github.com/nohwnd/Profiler/blob/main/csharp/Profiler/Tracer.cs#L148-L176>

Making reflection fast

- Figure out (slowly) where the info is once during setup:

<https://github.com/nohwnd/Profiler/blob/main/csharp/Profiler/Tracer.cs#L148-L176>

- Scrape it fast on every trace point:

<https://github.com/nohwnd/Profiler/blob/main/csharp/Profiler/Tracer.cs#L178-L191>

We have trigger, time, and all the details of the code we need 🎉

Tracing

ProfilerTracer : ITracer

- Save all info into a list.
- Optimize memory space by using structs.
- <https://github.com/nohwnd/Profiler/blob/main/csharp/Profiler/ProfilerTracer.cs#L68>

Index

- Each line knows its own index in the array
- It can easily refer to any other item, or item relative to it, e.g. next item

Index Text

```
0 $null = [Profiler.Tracer]::Patch($PSVersionTable ...  
1 $null = & $ScriptBlock  
2 {  
3 "Hello"
```

Timestamp

- Timestamp is start of a command
- Duration -> from start of this command, until start of the next command

Index	Text	Timestamp	Time
2	{	824365338456	0 ms
3	Start-Sleep -Seconds 1	824365338483	1001 ms
4	}	824375343557	0 ms

Level

- Each item knows how many items were on call stack when it run

Level	Text
-------	------

1	b
---	---

2	a
---	---

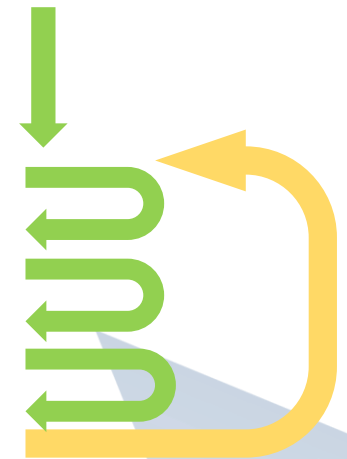
3	throw "uh oh!"
---	----------------

1	"ERR: \$_"
---	------------

```
1: try { b } catch { "ERR: $_" }
```

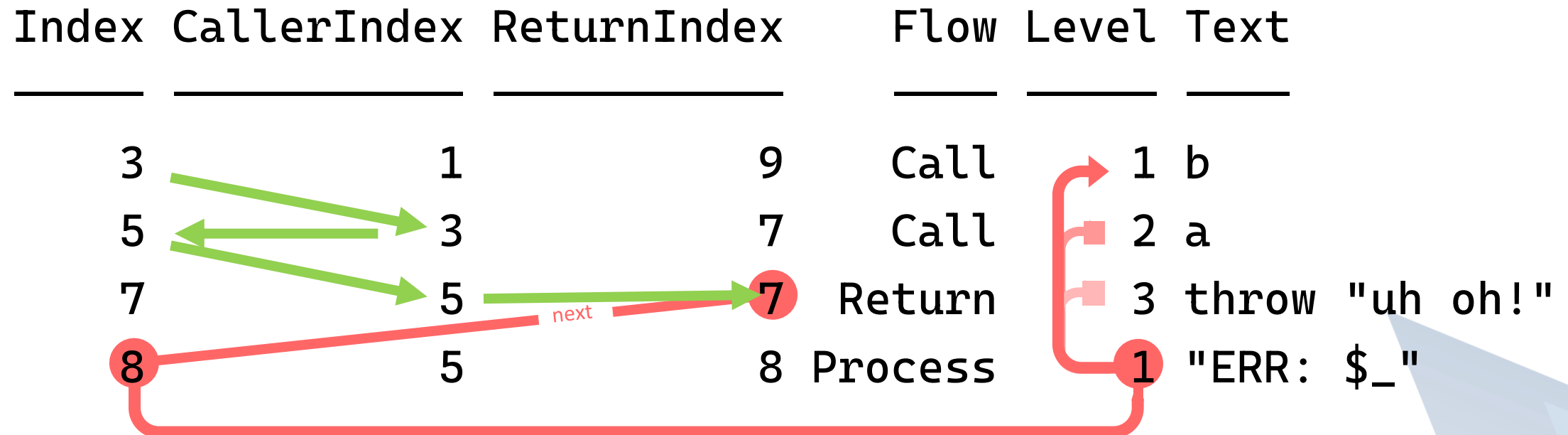
```
2: function b () { a }
```

```
3: function a () { throw "uh oh!" } ❌
```



CallerIndex, ReturnIndex, Flow

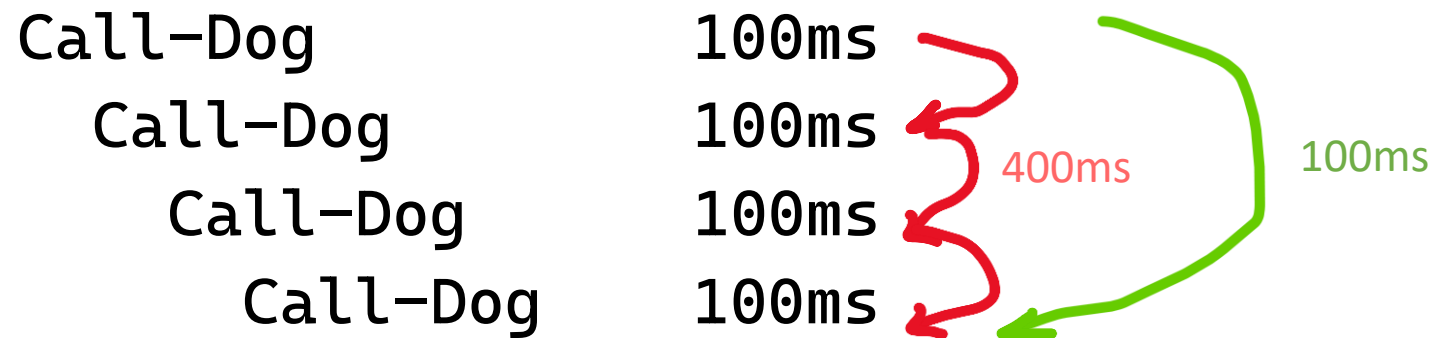
- By checking the next level, each item knows if it is a call, return, or process



[https://github.com/nohwnd/Profiler/blob/main/csharp/Profiler/Profiler ProcessFlow.cs](https://github.com/nohwnd/Profiler/blob/main/csharp/Profiler/Profiler%20ProcessFlow.cs)

Summarize data

- Add all the values together per line of code
- Be aware of recursive functions!



https://github.com/nohwnd/Profiler/blob/main/csharp/Profiler/Profiler_ProcessLines.cs

Recursion

- Duration in recursive functions is wrong!
- ✗ My 1st approach: Add all the durations of the same function together.
- ✓ Subtract time when we enter and leave the function.

Create final views

- Filter and sort all the summarized lines to get e.g. Top50SelfDuration.

<https://github.com/nohwnd/Profiler/blob/main/Profiler/Trace-Script.ps1#L204-L210>

Summary

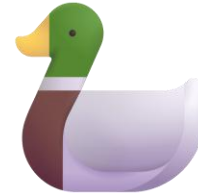
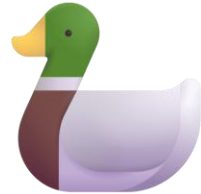
Don't get discouraged when things don't go as you planned. Dig deep, you might make a useful tool, or at least have fun learning!

Q&A

15 minutes



Bonus: Cooperating with Pester



<https://github.com/nohwnd/Profiler/blob/main/csharp/Profiler/Tracer.cs#L22-L27>

<https://github.com/nohwnd/Profiler/blob/main/csharp/Profiler/ProfilerTracer.cs>

<https://github.com/pester/Pester/blob/main/src/csharp/Pester/Tracing/CodeCoverageTracer.cs>

<https://github.com/nohwnd/Profiler/blob/main/csharp/Profiler/ITracer.cs#L12>

<https://github.com/pester/Pester/blob/main/src/csharp/Pester/Tracing/ITracer.cs>

<https://github.com/nohwnd/Profiler/blob/main/csharp/Profiler/ExternalTracerAdapter.cs#L44>