



Gold partner:



Generální partner:



Windows PowerShell 5 Preview April 2015

Michael Grafnetter
www.dsinternals.com

18. – 21. května 2015

Tech·Ed
DevCon 

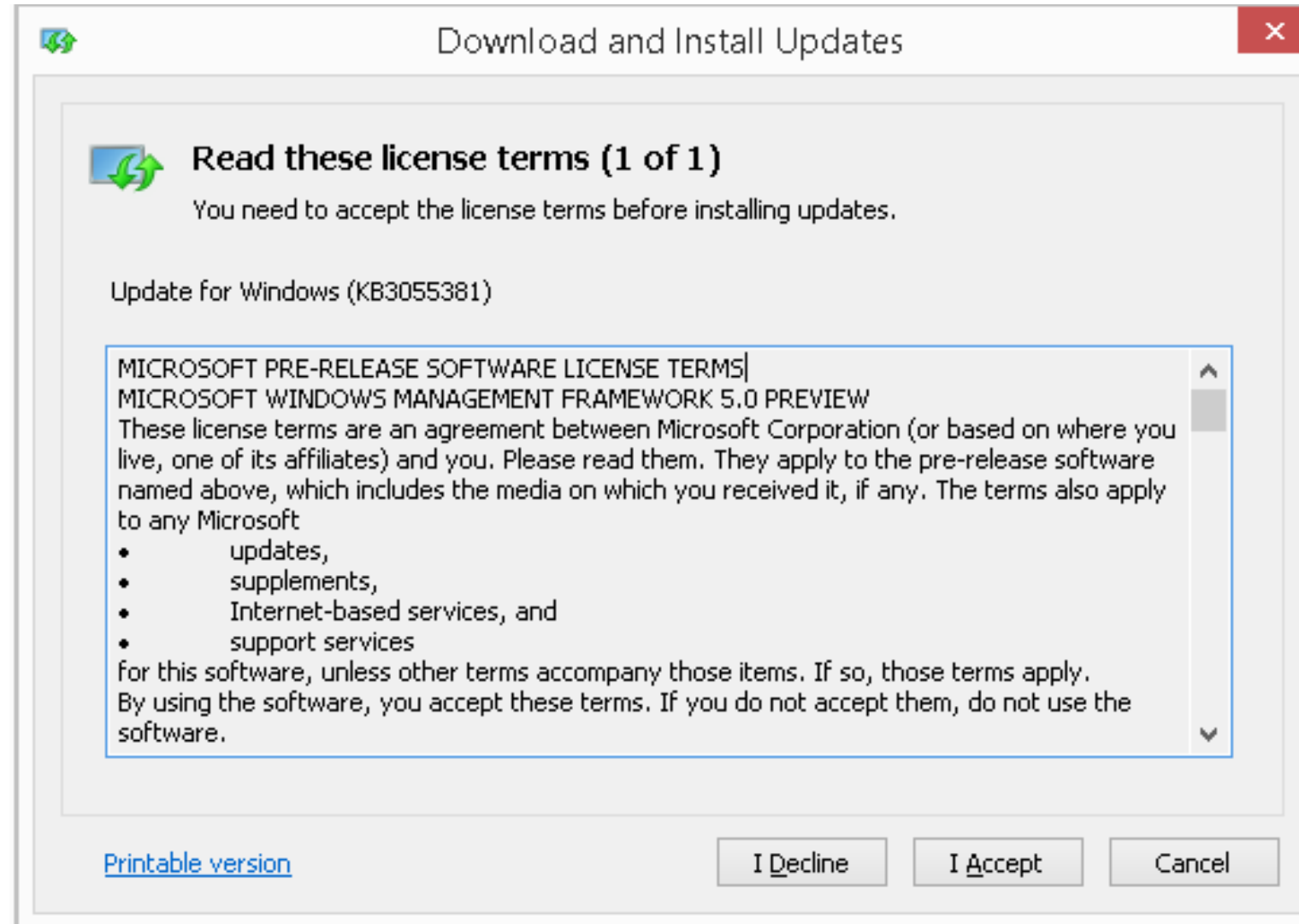
Agenda

- About PowerShell 4
- PowerShell ISE 4
- New Syntax
- Desired State Configuration
- New Cmdlets

Where to get it

- Built-in
 - Windows 10 Preview
- Windows Management Framework 5.0 Preview April 2015
 - Windows Server 2012 R2
 - Windows 8.1 Pro
 - Windows 8.1 Enterprise
 - Windows Server 2012
 - Windows 7 SP1
 - Windows Server 2008 R2 SP1

Installing PowerShell 5



Checking PowerShell Version

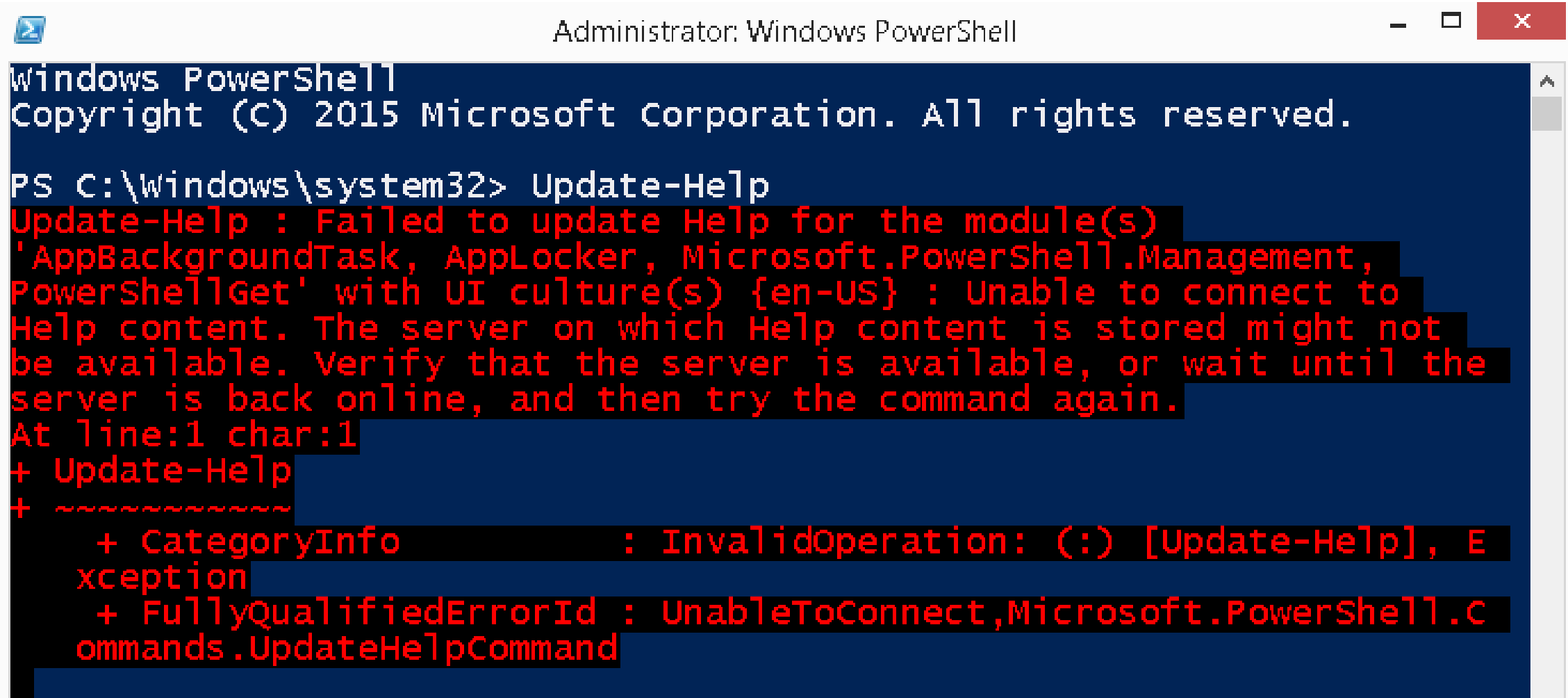
```
PS C:\> $PSVersionTable
```

Name	Value
----	-----
PSVersion	5.0.10105.0
WSManStackVersion	3.0
SerializationVersion	1.1.0.1
CLRVersion	4.0.30319.34003
BuildVersion	10.0.10105.0
PSCompatibleVersions	{1.0, 2.0, 3.0, 4.0...}
PSRemotingProtocolVersion	2.3

Known incompatibilities

- System Center 2012 Configuration Manager (without SP1)
- Windows Small Business Server 2011 Standard

Missing Documentation

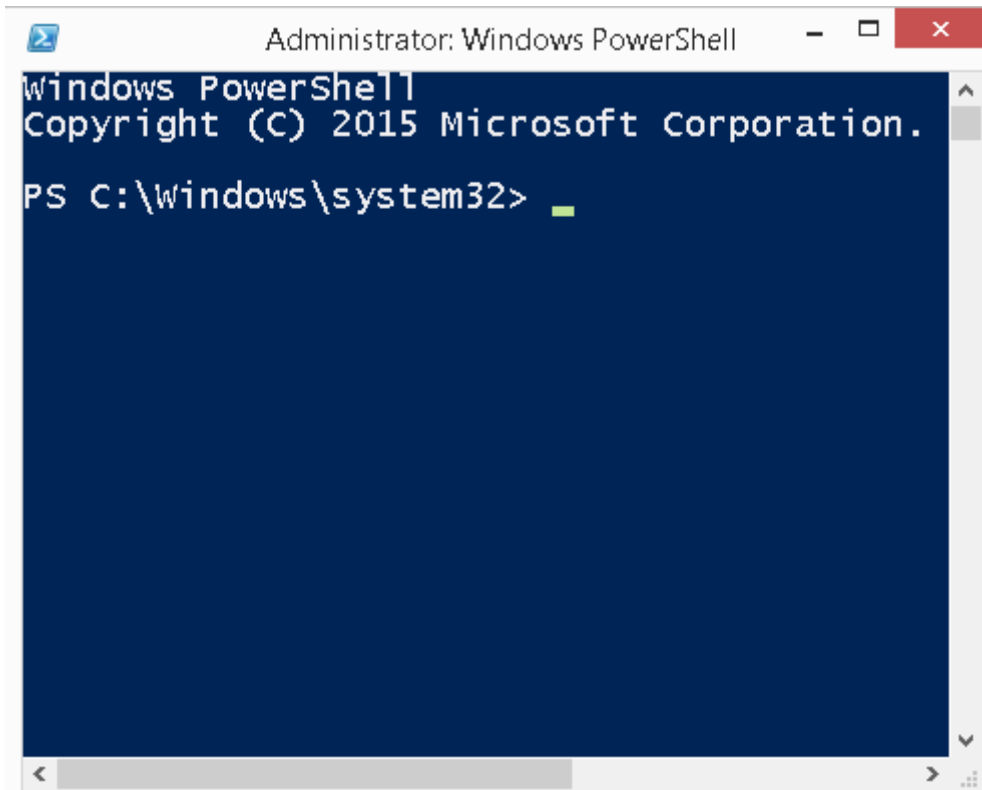


```
Administrator: Windows PowerShell

Windows PowerShell
Copyright (C) 2015 Microsoft Corporation. All rights reserved.

PS C:\Windows\system32> Update-Help
Update-Help : Failed to update Help for the module(s)
'AppBackgroundTask, AppLocker, Microsoft.PowerShell.Management,
PowerShellGet' with UI culture(s) {en-US} : Unable to connect to
Help content. The server on which Help content is stored might not
be available. Verify that the server is available, or wait until the
server is back online, and then try the command again.
At line:1 char:1
+ Update-Help
+ ~~~~~
+ CategoryInfo          : InvalidOperation: (:) [Update-Help], E
xception
+ FullyQualifiedErrorId : UnableToConnect,Microsoft.PowerShell.C
ommands.UpdateHelpCommand
```

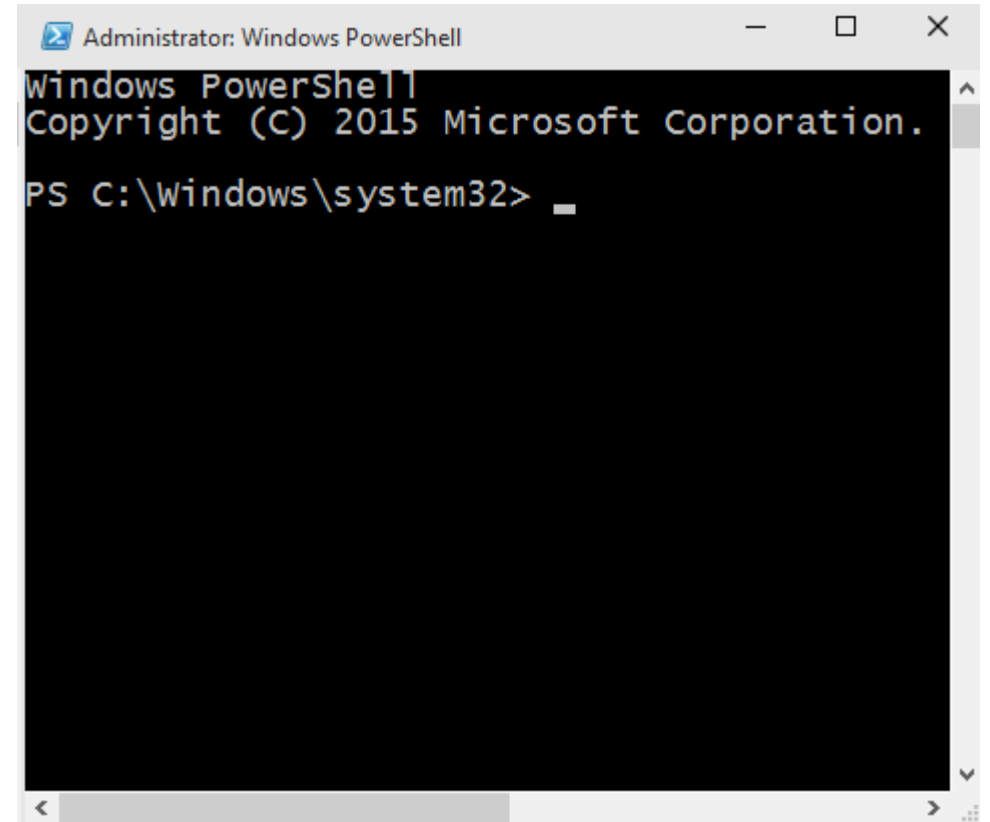
BLACK is BACK!



Administrator: Windows PowerShell

```
Windows PowerShell  
Copyright (C) 2015 Microsoft Corporation.  
  
PS C:\windows\system32> _
```

This image shows a Windows PowerShell window running as an administrator. The window has a blue background. The title bar reads "Administrator: Windows PowerShell". The command prompt displays the standard Windows PowerShell header, including the copyright notice for 2015 Microsoft Corporation, followed by the prompt "PS C:\windows\system32>" and a green cursor.

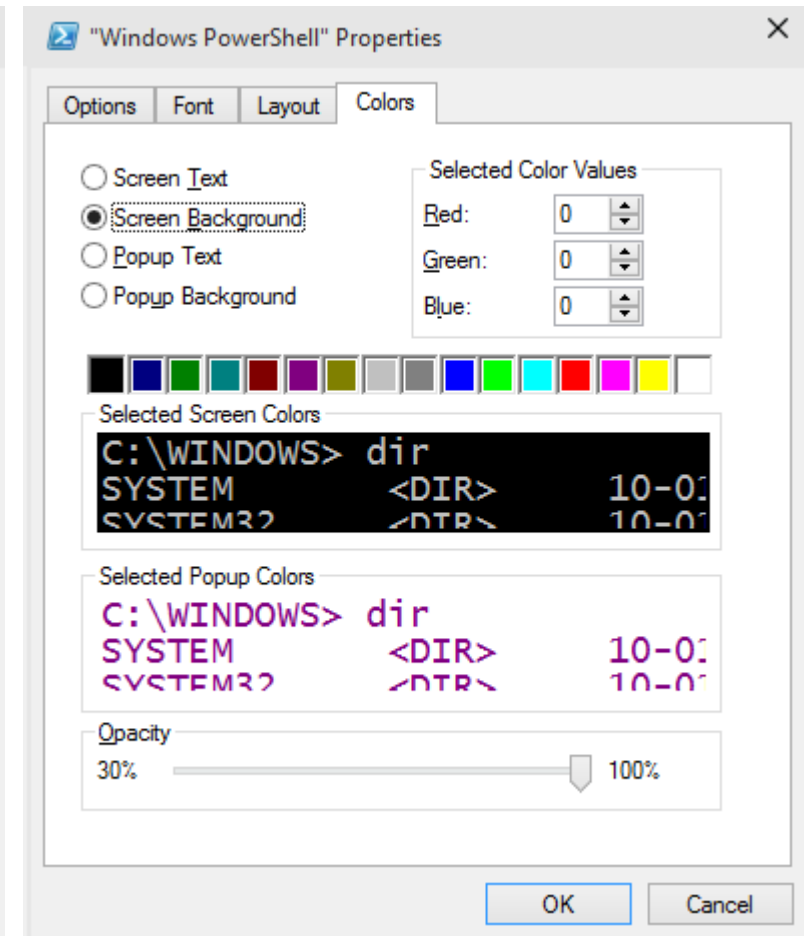
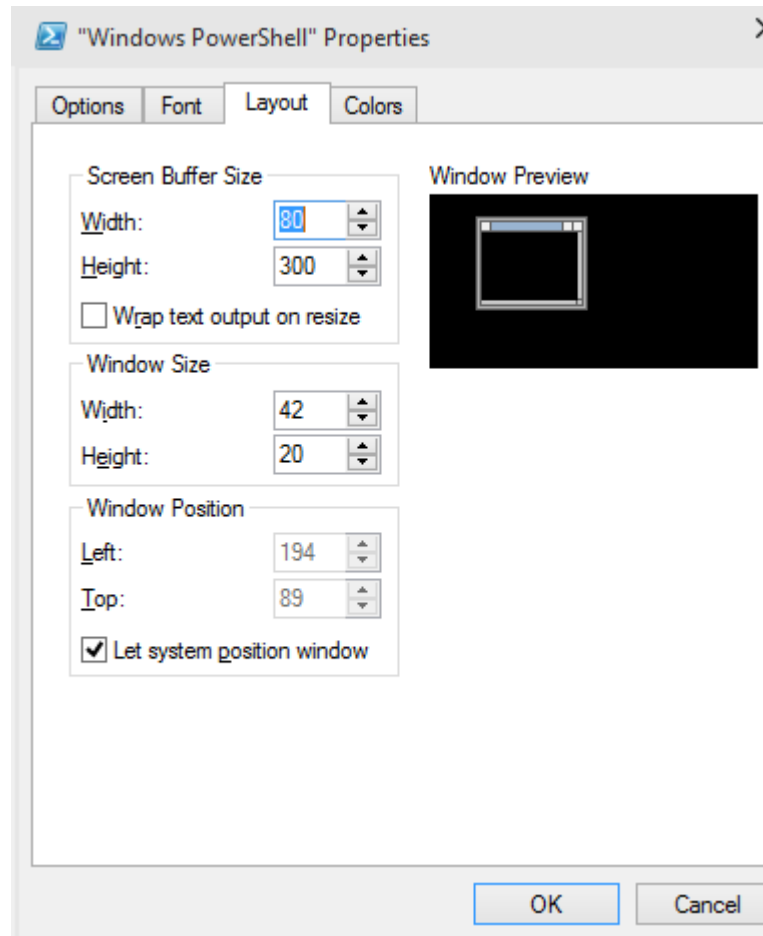
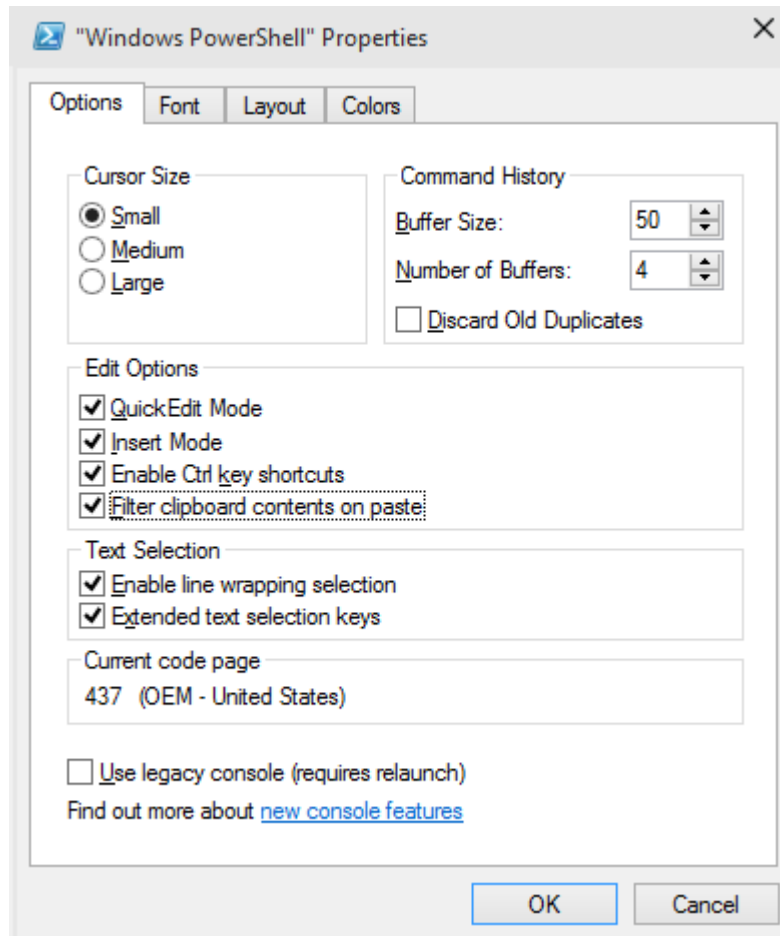


Administrator: Windows PowerShell

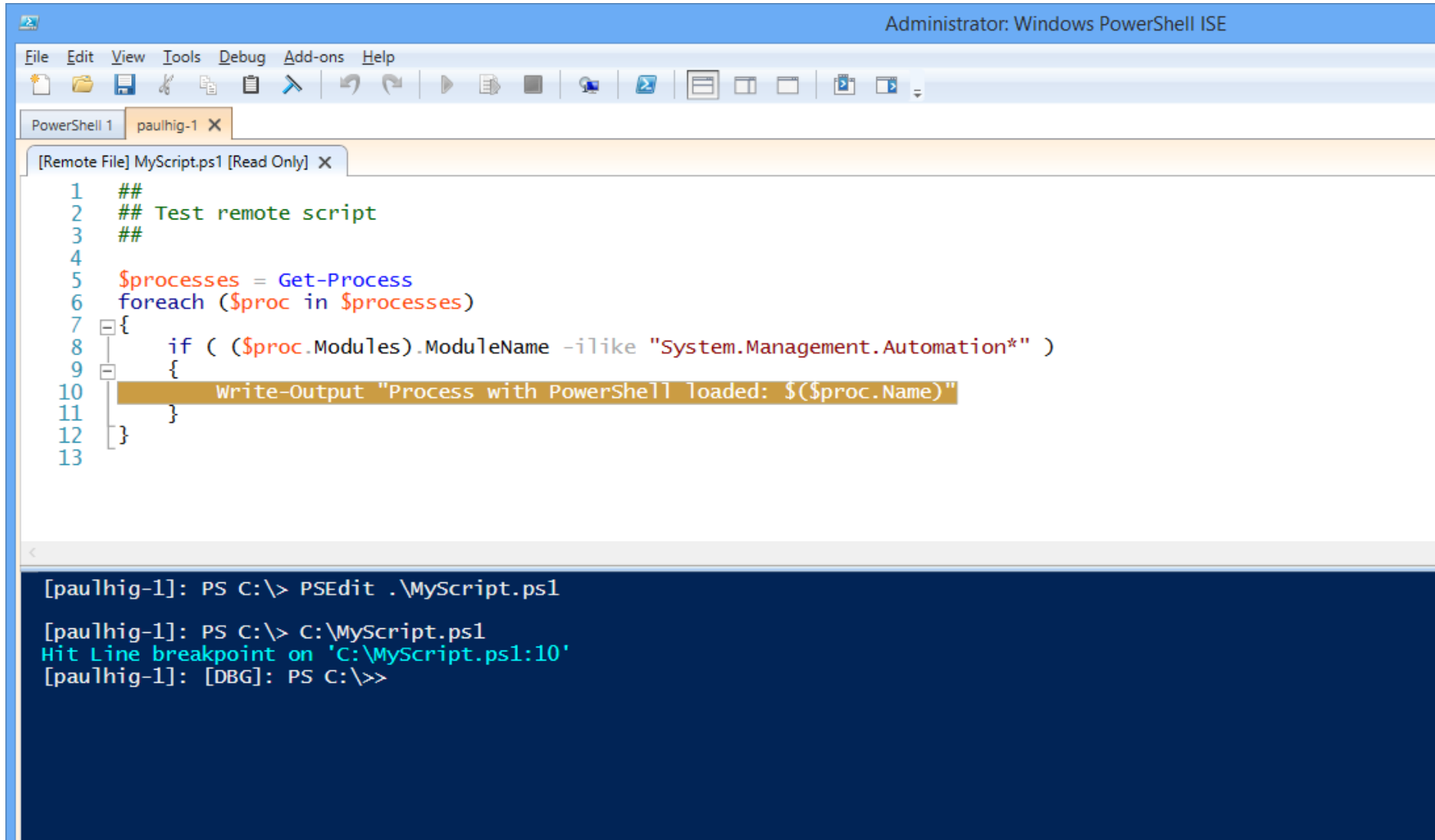
```
Windows PowerShell  
Copyright (C) 2015 Microsoft Corporation.  
  
PS C:\windows\system32> _
```

This image shows a Windows PowerShell window running as an administrator, similar to the one on the left but with a black background. The title bar reads "Administrator: Windows PowerShell". The command prompt displays the standard Windows PowerShell header, including the copyright notice for 2015 Microsoft Corporation, followed by the prompt "PS C:\windows\system32>" and a white cursor.

Windows 10 Command Line



ISE Remote Debugging



The screenshot displays the Windows PowerShell ISE interface. The title bar reads "Administrator: Windows PowerShell ISE". The menu bar includes File, Edit, View, Tools, Debug, Add-ons, and Help. The toolbar contains icons for file operations, editing, and execution. The script editor shows a file named "MyScript.ps1" with the following content:

```
1  ##
2  ## Test remote script
3  ##
4
5  $processes = Get-Process
6  foreach ($proc in $processes)
7  {
8      if ( ($proc.Modules).ModuleName -ilike "System.Management.Automation*" )
9      {
10         Write-Output "Process with PowerShell loaded: $($proc.Name)"
11     }
12 }
13
```

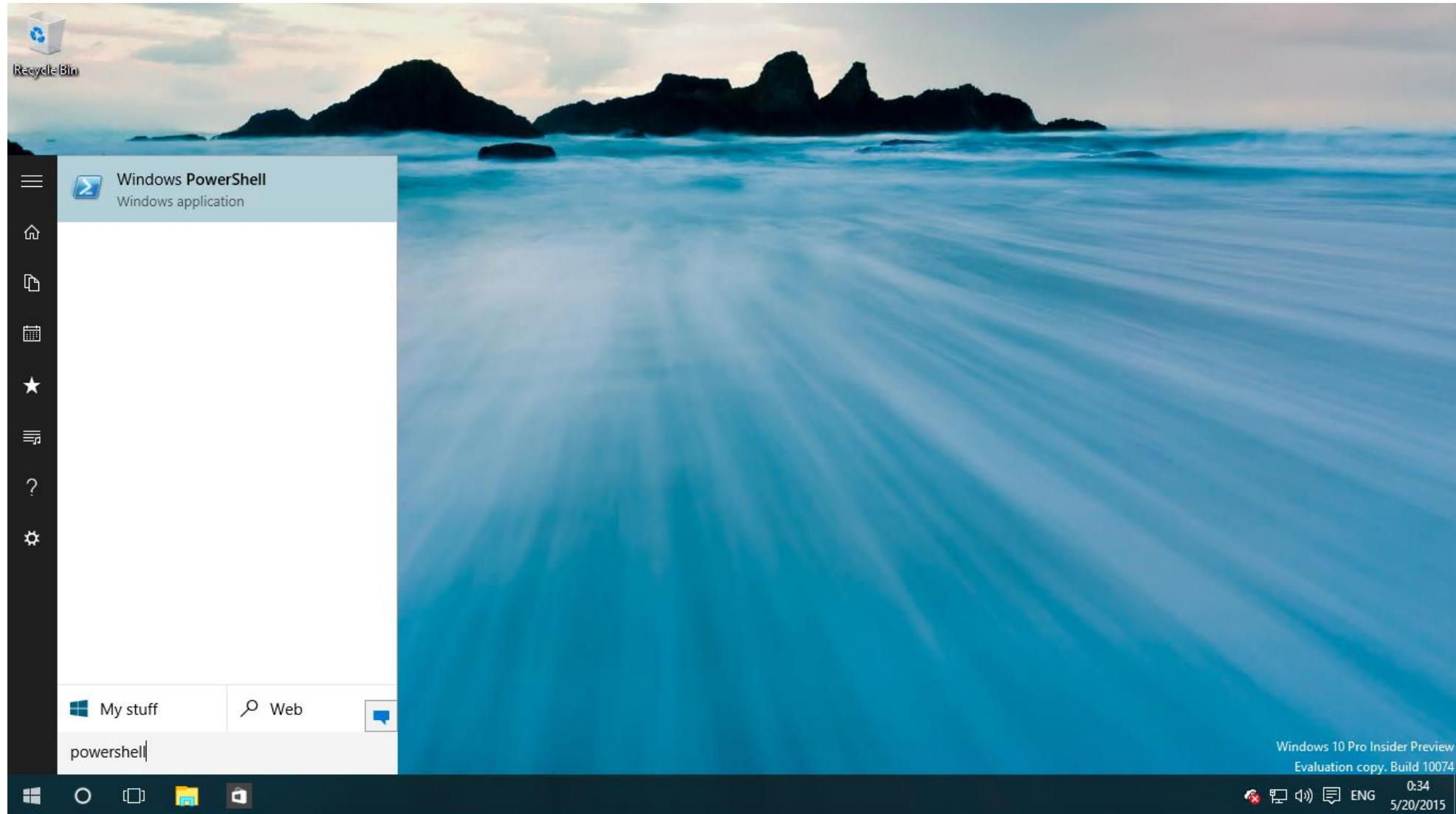
The console window at the bottom shows the execution of the script and the debugging process:

```
[paulhig-1]: PS C:\> PSEdit .\MyScript.ps1
[paulhig-1]: PS C:\> C:\MyScript.ps1
Hit Line breakpoint on 'C:\MyScript.ps1:10'
[paulhig-1]: [DBG]: PS C:\>>
```

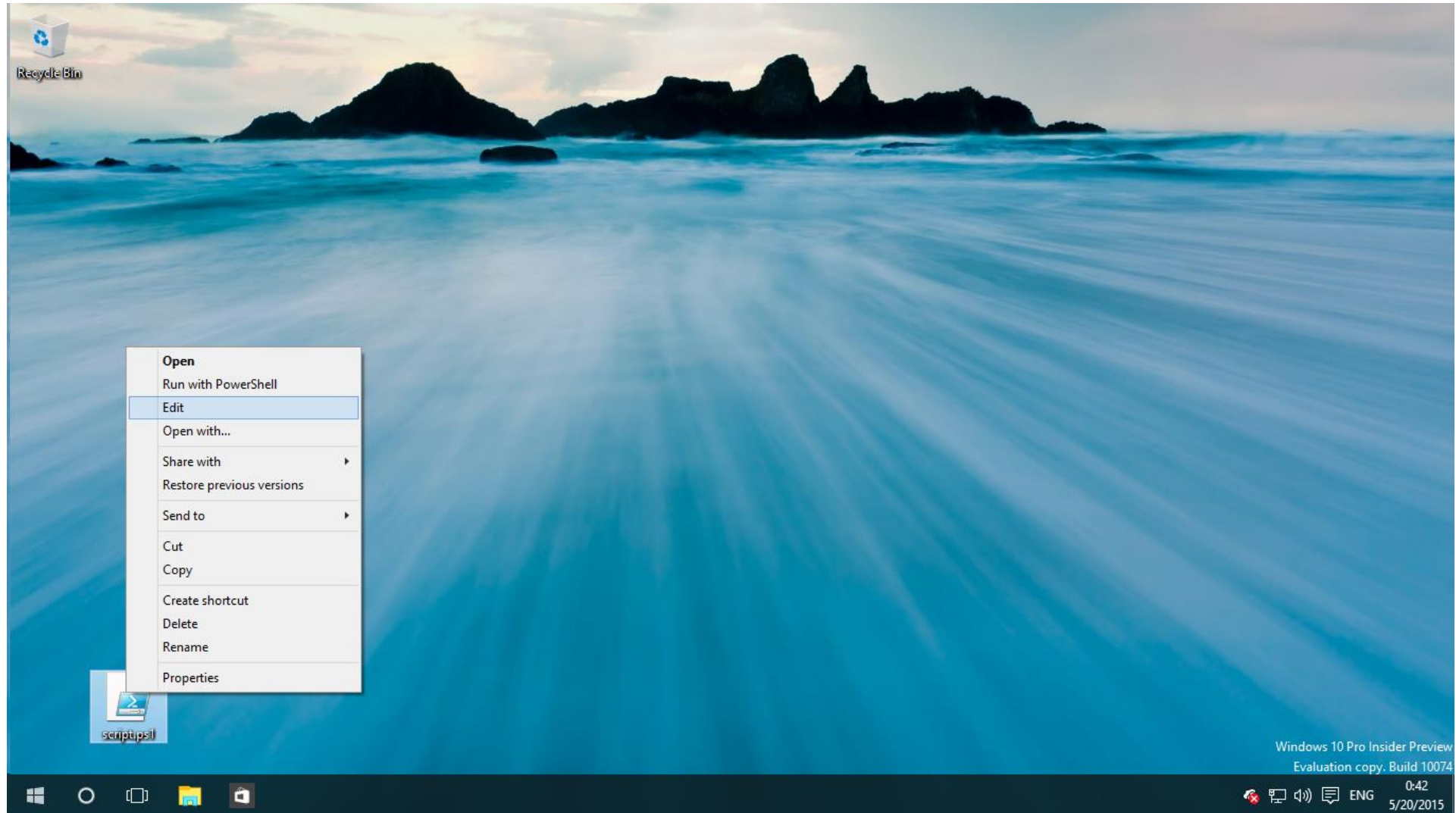
Advanced Debugging

- Attach to process
 - Enter-PSHostProcess
 - Exit-PSHostProcess
- Break All
 - Console: Ctrl+Break
 - ISE: Ctrl+B or Debug->Break All

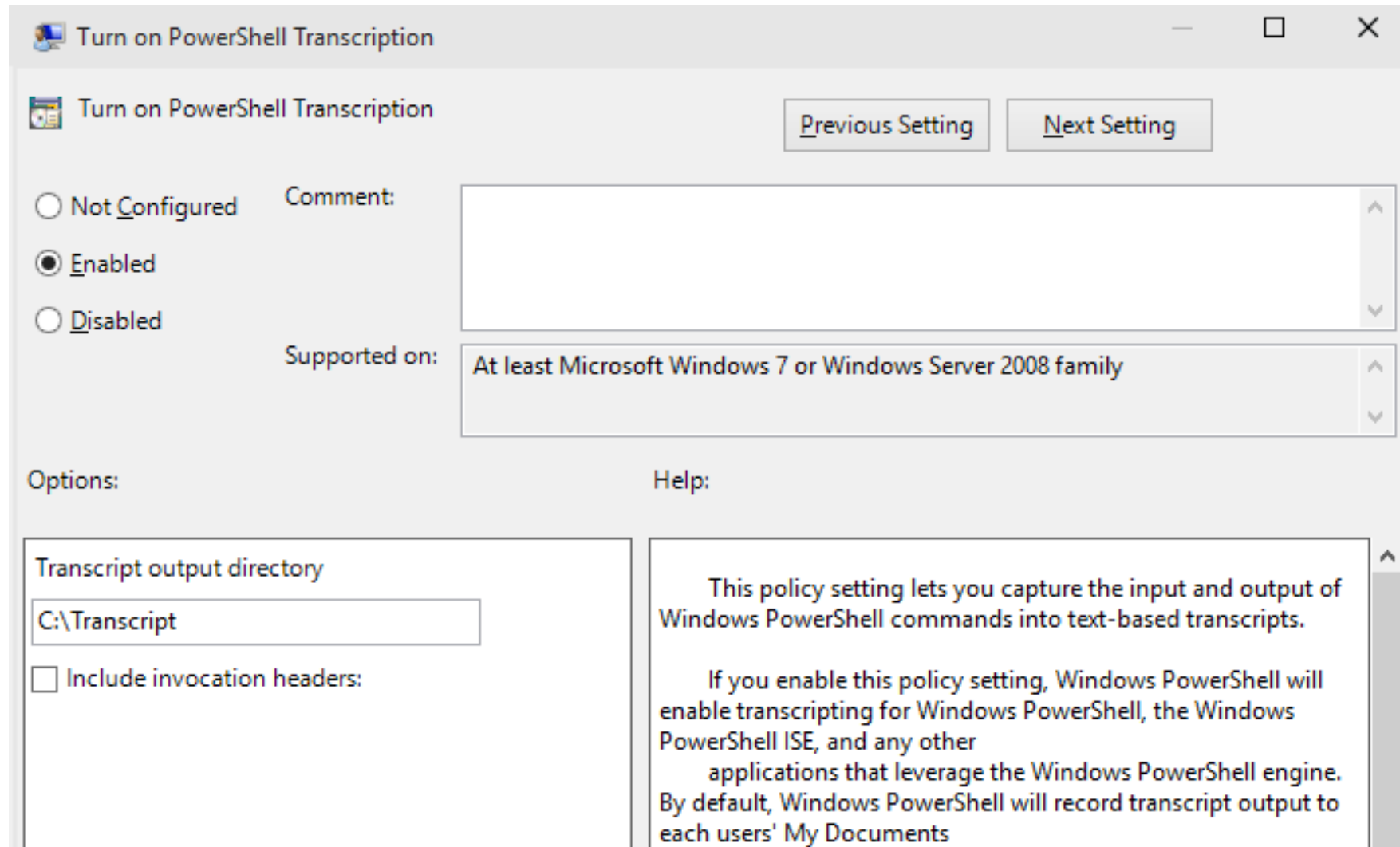
Cortana and ISE



Context Menu



PowerShell Transcription



The screenshot shows the 'Turn on PowerShell Transcription' window in Windows Settings. The window title is 'Turn on PowerShell Transcription'. Inside, there's a header with the same title and two buttons: 'Previous Setting' and 'Next Setting'. Below the header, there are three radio buttons for configuration: 'Not Configured', 'Enabled' (which is selected), and 'Disabled'. To the right of these is a 'Comment:' text box. Below the radio buttons is a 'Supported on:' section with a text box containing 'At least Microsoft Windows 7 or Windows Server 2008 family'. At the bottom, there are two sections: 'Options:' and 'Help:'. The 'Options:' section contains a 'Transcript output directory' text box with 'C:\Transcript' entered, and a checkbox labeled 'Include invocation headers:' which is currently unchecked. The 'Help:' section contains a scrollable area with text explaining the policy setting: 'This policy setting lets you capture the input and output of Windows PowerShell commands into text-based transcripts. If you enable this policy setting, Windows PowerShell will enable transcribing for Windows PowerShell, the Windows PowerShell ISE, and any other applications that leverage the Windows PowerShell engine. By default, Windows PowerShell will record transcript output to each users' My Documents'.

Turn on PowerShell Transcription

Turn on PowerShell Transcription

Previous Setting Next Setting

☐ Not Configured ☒ Enabled ☐ Disabled

Comment:

Supported on: At least Microsoft Windows 7 or Windows Server 2008 family

Options:

Transcript output directory

C:\Transcript

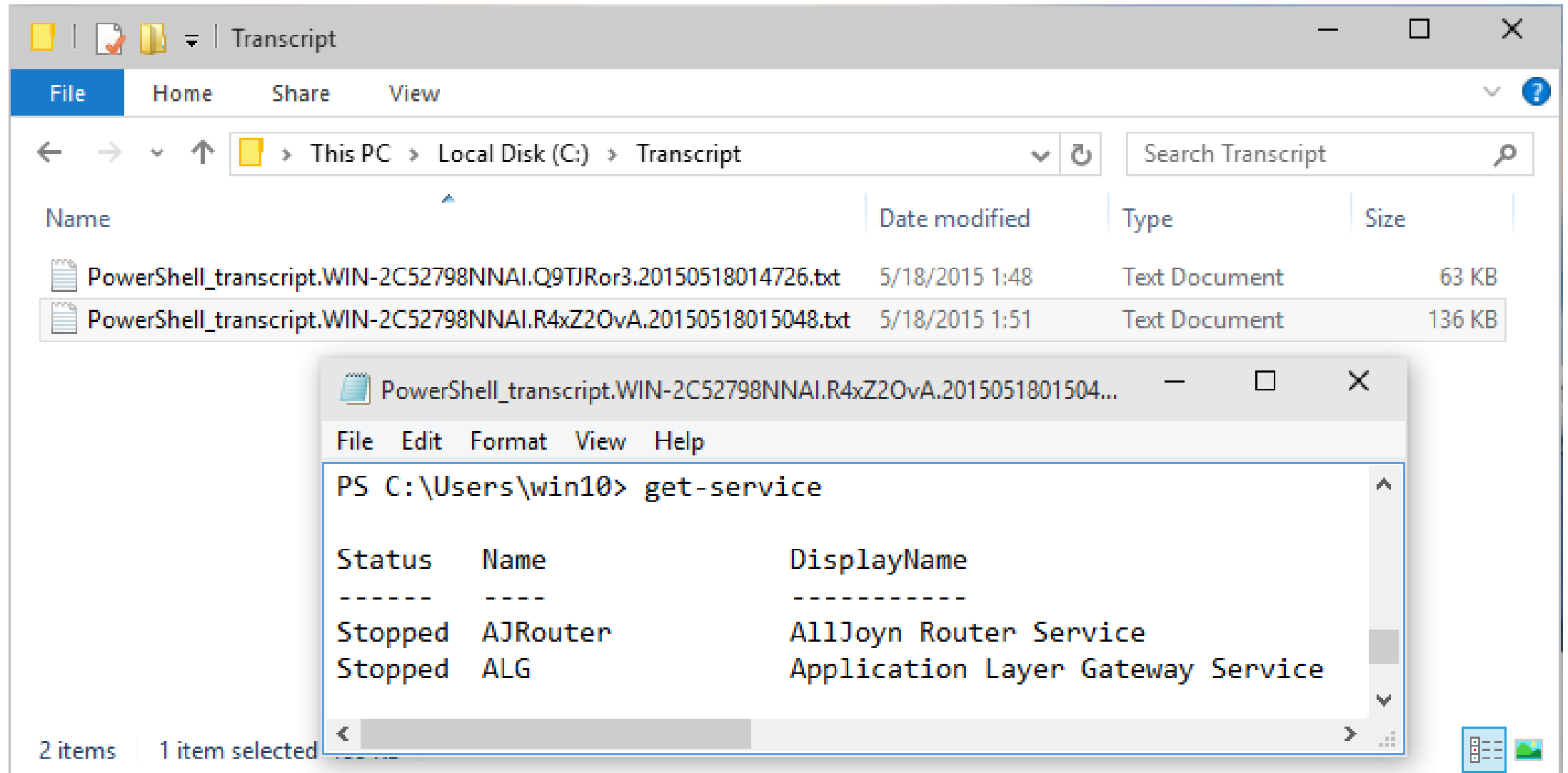
☐ Include invocation headers:

Help:

This policy setting lets you capture the input and output of Windows PowerShell commands into text-based transcripts.

If you enable this policy setting, Windows PowerShell will enable transcribing for Windows PowerShell, the Windows PowerShell ISE, and any other applications that leverage the Windows PowerShell engine. By default, Windows PowerShell will record transcript output to each users' My Documents

PowerShell Transcription



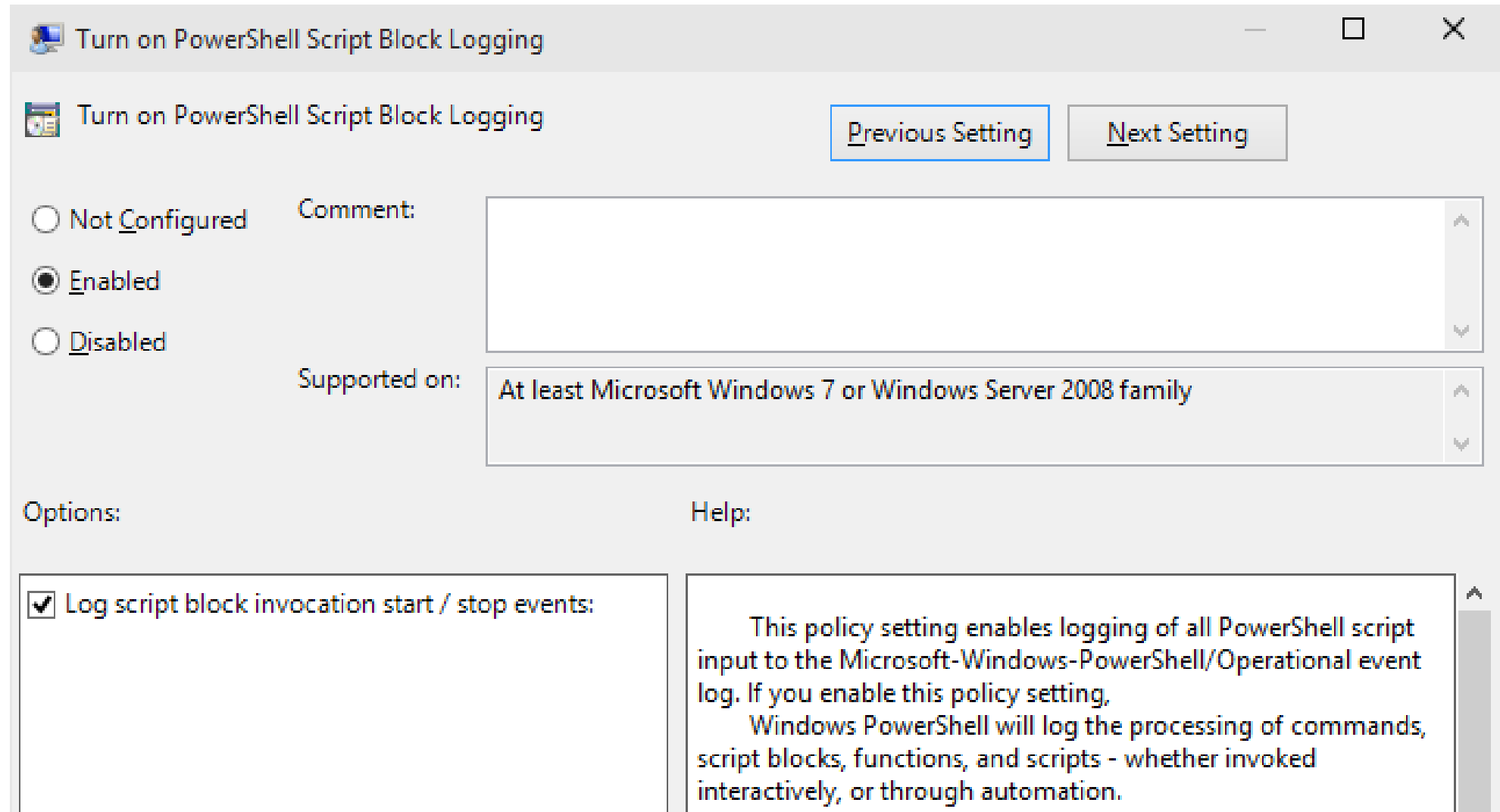
The screenshot displays a Windows File Explorer window titled 'Transcript' showing the contents of the 'Local Disk (C:) > Transcript' folder. Two text documents are listed:

Name	Date modified	Type	Size
PowerShell_transcript.WIN-2C52798NNAI.Q9TJRor3.20150518014726.txt	5/18/2015 1:48	Text Document	63 KB
PowerShell_transcript.WIN-2C52798NNAI.R4xZ2OvA.20150518015048.txt	5/18/2015 1:51	Text Document	136 KB

An overlaid PowerShell console window shows the command `get-service` being executed in the directory `C:\Users\win10>`. The output is as follows:

Status	Name	DisplayName
Stopped	AJRouter	AllJoyn Router Service
Stopped	ALG	Application Layer Gateway Service

PowerShell Script Block Logging



The screenshot shows the 'Turn on PowerShell Script Block Logging' window in Windows Settings. The window title is 'Turn on PowerShell Script Block Logging'. Inside, there's a header bar with the same title and two buttons: 'Previous Setting' and 'Next Setting'. Below the header, there are three radio buttons: 'Not Configured', 'Enabled' (which is selected), and 'Disabled'. To the right of these is a 'Comment:' text box. Below the radio buttons is a 'Supported on:' section with a text box containing 'At least Microsoft Windows 7 or Windows Server 2008 family'. At the bottom, there are two sections: 'Options:' and 'Help:'. The 'Options:' section has a checked checkbox for 'Log script block invocation start / stop events:'. The 'Help:' section contains a text box with the following text: 'This policy setting enables logging of all PowerShell script input to the Microsoft-Windows-PowerShell/Operational event log. If you enable this policy setting, Windows PowerShell will log the processing of commands, script blocks, functions, and scripts - whether invoked interactively, or through automation.'

Turn on PowerShell Script Block Logging

Turn on PowerShell Script Block Logging

Previous Setting Next Setting

☐ Not Configured ☒ Enabled ☐ Disabled

Comment:

Supported on: At least Microsoft Windows 7 or Windows Server 2008 family

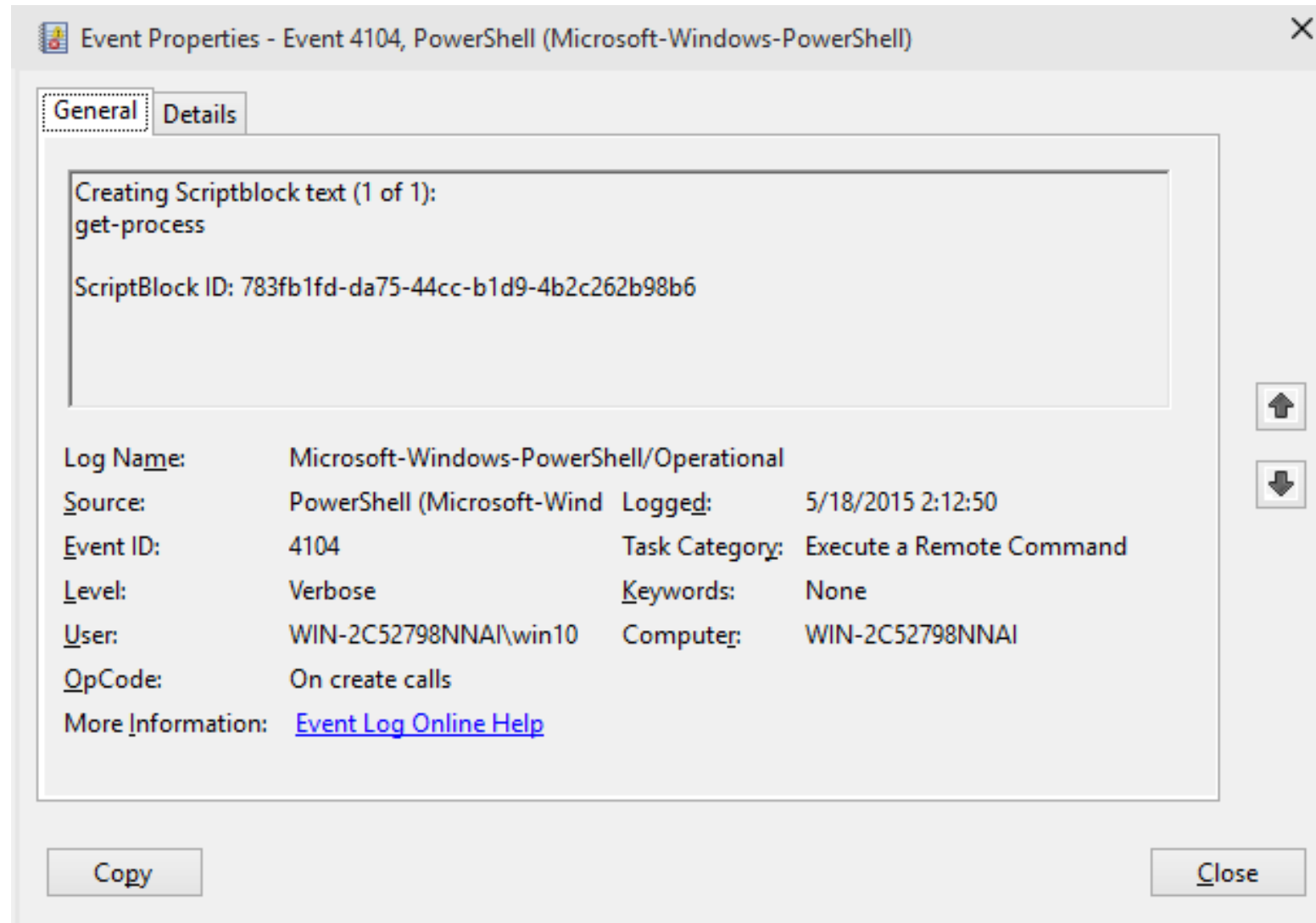
Options:

☒ Log script block invocation start / stop events:

Help:

This policy setting enables logging of all PowerShell script input to the Microsoft-Windows-PowerShell/Operational event log. If you enable this policy setting, Windows PowerShell will log the processing of commands, script blocks, functions, and scripts - whether invoked interactively, or through automation.

PowerShell Script Block Logging



Symlink and Hardlink Support

- **New-Item -ItemType SymbolicLink -Name MySymLinkDir -Target \$pshome**
- **New-Item -ItemType HardLink -Path C:\Temp -Name MyHardLinkFile.txt -Value \$pshome\profile.ps1**
- **New-Item -ItemType Junction -Path C:\Temp\MyJunctionDir -Value \$pshome**
- **Get-ChildItem**

DEMO

Managing .ZIP Files

- Compress-Archive

- [-Path] <String[]>

- [-DestinationPath] <String>

- [-CompressionLevel

- <Microsoft.PowerShell.Commands.CompressionLevel>]

- [-Update]

- Expand-Archive

- [-Path] <String>

- [-DestinationPath] <String>

DEMO

New-TemporaryFile

```
PS C:\> $tmp = New-TemporaryFile
```

```
PS C:\> $tmp.FullName
```

```
C:\Users\administrator.ADATUM\AppData\Local\Temp\tmp48FB.tmp
```

```
PS C:\> $tmp.Exists
```

```
True
```

```
PS C:\> $tmp = [System.IO.Path]::GetTempFileName()
```

```
PS C:\> $tmp
```

```
C:\Users\administrator.ADATUM\AppData\Local\Temp\tmpE868.tmp
```

Recursion Depth

```
PS C:\Users\Administrator> Get-ChildItem -Recurse -Depth 1
```

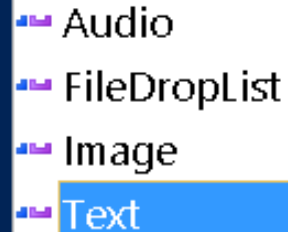
```
Directory: C:\Users\Administrator
```

Mode	LastWriteTime	Length	Name
----	-----	-----	----
d-r---	10/7/2013 1:55 PM		Contacts
d-r---	10/7/2013 2:23 PM		Desktop
d-r---	10/7/2013 1:55 PM		Documents
d-r---	10/7/2013 1:55 PM		Downloads
d-r---	10/7/2013 2:16 PM		Favorites

Clipboard Manipulation

```
PS C:\> Get-Clipboard -Format Image
```

```
Tag                :  
PhysicalDimension  : {width=871, Height=321}  
Size               : {width=871, Height=321}  
Width              : 871  
Height             : 321  
HorizontalResolution : 96  
VerticalResolution  : 96  
Flags              : 335888  
RawFormat           : [ImageFormat: b96b3caa-0728-11d3-9d7b-0000f81ef32e]  
PixelFormat         : Format32bppRgb  
Palette             : System.Drawing.Imaging.ColorPalette  
FrameDimensionsList : {7462dc86-6180-4c7e-8e3f-ee7333a7a483}  
PropertyIdList      : {}  
PropertyItems       : {}
```



- Audio
- FileDropList
- Image
- Text

```
PS C:\> Get-Clipboard -Format
```


DEMO

New-Guid

```
PS C:\> New-Guid
```

```
Guid
```

```
----
```

```
de31f4bd-7279-4f93-950d-cf77a1502141
```

```
PS C:\> [guid]::NewGuid()
```

```
Guid
```

```
----
```

```
7b6286e3-9111-44c6-83ed-6d88415fc7c5
```

Convert-String

```
PS C:\> "Lee Holmes", "Steve Lee", "Jeffrey Snover" |  
    Convert-String -Example @{Before = 'Bill Gates'; After = 'Gates, B.'}  
Holmes, L.  
Lee, S.  
Snover, J.
```

DEMO

ConvertFrom-String

```
>> Get-Content .\addresses.output.txt | ConvertFrom-String -TemplateFile  
.\addresses.template.txt |  
>>> Format-Table -Auto
```

<u>ExtentText</u>	Name	City	State
-----	----	----	-----
Ana Trujillo...	Ana Trujillo	Redmond	WA
Antonio Moreno...	Antonio Moreno	<u>Renton</u>	WA
Thomas Hardy...	Thomas Hardy	Seattle	<u>WA</u>
Christina Berglund...	Christina Berglund	Redmond	<u>WA</u>
Hanna Moos...	Hanna Moos	Puyallup	WA

DEMO

Microsoft Research FlashExtract

FlashExtract: A Framework for Data Extraction by Examples

Wu L²
extremely or completely at least
available to the public

Sarmit Chakraborty
sarmit@cs.cmu.edu
sachin@cs.cmu.edu

Acknowledgments

rather than the other, that combine models and data (e.g., Bell & Waples, 1995; Hightower, 1996). In order to compare multiple life-history data, this model is divided into three data sets or further subdivisions or quantifying. We present a general framework presented in several previous data sets and re-estimated life-history using examples. It includes an introductory model for life-history data, a model for life-history data, and a model for life-history data. We present a general framework presented in several previous data sets and re-estimated life-history using examples. It includes an introductory model for life-history data, a model for life-history data, and a model for life-history data. We present a general framework presented in several previous data sets and re-estimated life-history using examples. It includes an introductory model for life-history data, a model for life-history data, and a model for life-history data.

Categories and Subject Descriptors: D.1.2 [Programming Techniques]: Automatic programming; D.2.2 [Analysis and Synthesis]: Program synthesis

General Terms: languages, algorithms, neural networks

Keywords: program synthesis, end-user programming, program synthesis by example

1. Introduction

information is available in documents or various types such as text, tables, spreadsheets, and drawings. These documents are their creators' gold. Security is strong and managing hierarchical data by combining rows and columns is a key to the ability to find, modify, insert, and delete information. But to extract the underlying data or even to look such as data processing, sorting, filtering, and summarizing rows, or transforming data is another level of complexity.

^a Work done during the intervention at Microsoft Research.

[illegible]

video and to demonstrate it to various domain experts. It encourages non-domain experts, including languages like REX, APL, which have been designed to support string processing to test them, which have reasons to believe that string (and array) data is not machine using a rich built-in library of string and numerical functions, or to use the already written in visual based text programming languages, with mathematical data types, *etc.*, as it can be used to extract data from web pages, but also for the additional benefit of knowing the underlying concepts of a database.

relying on programming solutions to data abstraction have time and space complexities, first, the complexity are domain specific and second, knowledge engineering is a domain technique to solve hard decision types, second, they require some thinking in the data underlying the problem, third, they are not suitable for the data underlying the problem, and last, they are not suitable for the data underlying the problem, and last, they are not suitable for the data underlying the problem.

In this paper, we address the problems of developing a uniform and user-friendly interface to support data extraction from semi-structured documents at various layers. Our methodology includes two key novel aspects: a uniform user interaction model across different document types, and a generic inductive program synthesis methodology.

Diagram and User-Proxy Interaction Model Our interaction engine supports data abstraction via diagrams. The user interacts with the system by providing a textual hierarchical definition of the data. The system works by interpreting structured data as sequence constructs, the user can provide data in the form of the sequence constructs, or the user can provide data in the form of the value data results and the user is automatically with each other. An interesting aspect is that this model is independent of the underlying document type. This is based on our observation that domain discourse of type data can be found in a common 1-dimentional presentation style. we allow users to provide attempts by highlighting two-dimensional data results. The system automatically suggests the value data results that the user wants to extract or transform into hierarchical structures around the data.

Inductive Program Synthesis Framework To enable data extraction from samples, we average inductive program synthesis to find candidate scripts from samples in an existing high domain-specific language (DSL), the key technical contribution of this paper is an inductive program synthesis framework that uses any high-level or inductive synthesizers from any definition of DSL to find data extraction scripts across domains (Figure 2). We describe an expressive DSL for capturing data extraction rules, make program synthesis more efficient by using a heuristic search, and make semantic debugging methods applicable to tasks learned by the

[illegible]

```

function RegionToRegionProg (R0::Region, Region:: Q)
  Local Prog:: W
12  
$$Q' := \{ (R_1 \sqcup R_2, R') \mid (R_1, R') \in Q \}$$

13  return N.Locals(Q') // no more applied, the region is
function N.Locals(R0::Region, T:: Q) Local Prog:: W
14  
$$P := \emptyset$$

15  foreach C in N.AllNodes
16    
$$P := P \cup C.Locals(Q)$$

17  return P

```

```

function Rep_Layers(Net(Phi), Layers, Init(Phi), q) : List(Prog) is
  // Let P and X be the functions that compute
  // the support of Rep_i of
  //
  // 1.  $q_1 = \text{Rep}_1(\Phi_1, \Phi_2, \Phi_3)$  // Find minimum X
  // 2.  $X_1 \leftarrow \text{Layers}(\text{support}(\text{Rep}_1, \Phi_1))$ 
  // 3.  $q_2 = \text{Rep}_2(\Phi_1, \Phi_2, \Phi_3)$  //  $q_2(1) \in \text{supp}(X_1), 1 \leq 2 \leq m$ 
  // 4.  $P_1 \leftarrow \text{Layers}(\Phi_1)$  // Layers Rep's function P
  // 5.  $q_3 = \text{Rep}_3(\Phi_1, \Phi_2, \Phi_3)$  //  $q_3(1) \in \text{supp}(P_1), 1 \leq 2 \leq m$ 
  // 6.  $P_2 \leftarrow \text{Layers}(\Phi_2)$  // Layers Rep's support P
  // 7.

```

```

17  return  $\bigcup_{i=1}^n \text{Bingo}(\mathcal{P}_i, \mathcal{P}_i, \text{do})$ 
18   $\left[ \begin{array}{l} \text{Bingo}(\mathcal{P}_i, \mathcal{P}_i, \text{do}) \\ \mathcal{P}_i \leftarrow \mathcal{P}_i \cup \text{Bingo}(\mathcal{P}_i, \mathcal{P}_i, \text{do}) \end{array} \right]$ 
19  return  $\bigcup_{i=1}^n \text{Bingo}(\mathcal{P}_i, \mathcal{P}_i, \text{do})$ 

```

function **Bingo**: $\text{Bingo}(\text{Bingo}(\mathcal{P}_i, \mathcal{P}_i, \text{do}), \text{do}) \rightarrow \text{Bingo}(\mathcal{P}_i, \mathcal{P}_i, \text{do})$

1. Let \mathcal{P}_i be the non-trivial class from the expression of **Bingo**. \mathcal{P}_i

2. Let $\mathcal{P}_i \leftarrow \mathcal{P}_i \cup \text{Bingo}(\mathcal{P}_i, \mathcal{P}_i, \text{do})$

3. Let \mathcal{P}_i include all possible instances of assignment \mathcal{P}_i

1. $X := \{Q^2 \in \mathbb{Q}^2 \mid Q^2 = (2p_1, 2p_2), p_1, p_2 \in \mathbb{Q}\}$
2. $W := \{Q^2 \in \mathbb{Q}^2 \mid Q^2 = (2p_1, 2p_2), A \text{ divides } (Q^2) \neq \emptyset\}$
3. $V := \{Q^2 \in \mathbb{Q}^2 \mid Q^2 = (2p_1, 2p_2), A \text{ divides } (Q^2) \neq \emptyset\}$
4. $T := \{Q^2 \in \mathbb{Q}^2 \mid Q^2 = (2p_1, 2p_2), A \text{ divides } (Q^2) \neq \emptyset\}$
5. $T := \{Q^2 \in \mathbb{Q}^2 \mid Q^2 = (2p_1, 2p_2), A \text{ divides } (Q^2) \neq \emptyset\}$
6. $T := \{Q^2 \in \mathbb{Q}^2 \mid Q^2 = (2p_1, 2p_2), A \text{ divides } (Q^2) \neq \emptyset\}$
7. $T := \{Q^2 \in \mathbb{Q}^2 \mid Q^2 = (2p_1, 2p_2), A \text{ divides } (Q^2) \neq \emptyset\}$
8. $T := \{Q^2 \in \mathbb{Q}^2 \mid Q^2 = (2p_1, 2p_2), A \text{ divides } (Q^2) \neq \emptyset\}$
9. $T := \{Q^2 \in \mathbb{Q}^2 \mid Q^2 = (2p_1, 2p_2), A \text{ divides } (Q^2) \neq \emptyset\}$
10. $T := \{Q^2 \in \mathbb{Q}^2 \mid Q^2 = (2p_1, 2p_2), A \text{ divides } (Q^2) \neq \emptyset\}$

```

1   $P \leftarrow P \cup \{ \text{Neighbours}(P_1, \dots, P_k) \mid 1 \leq i \leq n, P_i \in P \}$ 
2  return  $\text{Quantity}(P, Q)$ 

```

Figure 4: *VerbalizeInductive*

```

function FilzSeriesList : LSeries (G:GQ, F:Finite, L:List(T)) : GQ :=
  LSeries(F)begin
    /- let G and F be the group and sequence
    /- sequence of FilzSeries. -/
    R1 := R.LSeries(G) /- let R series R
    Q1 := [(0,F)(0,F)] : GQ, T := GQ, 0 ≤ i ≤ LLen(Q) - 1
    R2 := R.LSeries(Q1) /- let R series R
    R := []
    for i in 1 to R1 do
      for j in 1 to R2 do
        R := R + [(i,j)] : GQ, T
      end
    end
  end

```

```

function FilSeries(Lamers: (Set)(FilSeries, LSet)(T)) (Q):
  LSet (F) do
    /- Lam F is the sequence segment of FilSeries. n
    lam(Q) is (x1, x2) + Q /-
    
$$F_i := L\text{amers}(Q)$$

    /- lamers sequence F
    
$$F := \emptyset$$

    Iterate  $F_i \in F_i$  do
      lamers := F
      lamers := Q
      F := F + lamers

```

```

57  $\mathcal{Z}_T := [P]_T$ 
58  $\text{inits} \leftarrow \text{Inits}(\text{inits}, \text{inits of } (\mathcal{Z}_T, Y_T, \mathcal{R}))$ 
59  $\text{inits} \leftarrow 0 \dots |\mathcal{Z}_T| - 1$  do
60    $i \leftarrow \text{inits of } (\mathcal{Z}_T, Y_T) + 1 \mid \text{inits of } (\mathcal{Z}_T, Y_T)$ 
61   if  $\text{clear} \leftarrow \text{Clear}(\text{inits} \leftarrow i)$ 
62     else  $\text{inits} \leftarrow \text{C22}(i \text{ inits}, i)$ 
63   if  $\text{clear} \leftarrow \text{Clear}(\text{inits} \leftarrow i)$ 
64      $\beta \leftarrow \beta \cup \text{Inits}(\text{inits}, \text{inits}, P)$ 
65    $\text{inits} \leftarrow \text{Clear}(\beta, i)$ 

```

```

17   $q_i := (|x_1, y_1, w_1|, |x_2, y_2, w_2|)$ 
18   $q_i := (|x_1, y_1, w_1|, |x_2, y_2, w_2|)$ 
19   $P_i := A \cdot \text{Lattice}(q_i)$ 
20   $P_i := B \cdot \text{Lattice}(q_i)$ 
21  if  $P_i = B$  or  $P_i = B$  then return  $i$ 
22  return  $i$ 
23  foreach  $P_i \in P$  do
24    foreach  $P_j \in P$  do
25       $P_i \leftarrow P_i \cup P_j$ 

```

[illegible]

```

10  if (P == 0) return 0;
11  if (P == 1) return 1;
12  if (P == 2) return 2;
13  if (P == 3) return 4;
14  if (P == 4) return 8;
15  if (P == 5) return 16;
16  if (P == 6) return 32;
17  if (P == 7) return 64;
18  if (P == 8) return 128;
19  if (P == 9) return 256;
20  if (P == 10) return 512;
21  if (P == 11) return 1024;
22  if (P == 12) return 2048;
23  if (P == 13) return 4096;
24  if (P == 14) return 8192;
25  if (P == 15) return 16384;
26  if (P == 16) return 32768;
27  if (P == 17) return 65536;
28  if (P == 18) return 131072;
29  if (P == 19) return 262144;
30  if (P == 20) return 524288;
31  if (P == 21) return 1048576;
32  if (P == 22) return 2097152;
33  if (P == 23) return 4194304;
34  if (P == 24) return 8388608;
35  if (P == 25) return 16777216;
36  if (P == 26) return 33554432;
37  if (P == 27) return 67108864;
38  if (P == 28) return 134217728;
39  if (P == 29) return 268435456;
40  if (P == 30) return 536870912;
41  if (P == 31) return 1073741824;
42  if (P == 32) return 2147483648;
43  if (P == 33) return 4294967296;
44  if (P == 34) return 8589934592;
45  if (P == 35) return 17179869184;
46  if (P == 36) return 34359738368;
47  if (P == 37) return 68719476736;
48  if (P == 38) return 137438953472;
49  if (P == 39) return 274877906944;
50  if (P == 40) return 549755813888;
51  if (P == 41) return 1099511627776;
52  if (P == 42) return 2199023255552;
53  if (P == 43) return 4398046511104;
54  if (P == 44) return 8796093022208;
55  if (P == 45) return 17592186044416;
56  if (P == 46) return 35184372088832;
57  if (P == 47) return 70368744177664;
58  if (P == 48) return 140737488355328;
59  if (P == 49) return 281474976710656;
60  if (P == 50) return 562949953421312;
61  if (P == 51) return 1125899906842624;
62  if (P == 52) return 2251799813685248;
63  if (P == 53) return 4503599627370496;
64  if (P == 54) return 9007199254740992;
65  if (P == 55) return 18014398509481984;
66  if (P == 56) return 36028797018963968;
67  if (P == 57) return 72057594037927936;
68  if (P == 58) return 144115188075855872;
69  if (P == 59) return 288230376151711744;
70  if (P == 60) return 576460752303423488;
71  if (P == 61) return 1152921504606846976;
72  if (P == 62) return 2305843009213693952;
73  if (P == 63) return 4611686018427387904;
74  if (P == 64) return 9223372036854775808;
75  if (P == 65) return 18446744073709551616;
76  if (P == 66) return 36893488147419103232;
77  if (P == 67) return 73786976294838206464;
78  if (P == 68) return 147573952589676412928;
79  if (P == 69) return 295147905179352825856;
80  if (P == 70) return 590295810358705651712;
81  if (P == 71) return 1180591620717411303424;
82  if (P == 72) return 2361183241434822606848;
83  if (P == 73) return 4722366482869645213696;
84  if (P == 74) return 9444732965739290427392;
85  if (P == 75) return 18889465931478580854784;
86  if (P == 76) return 37778931862957161709568;
87  if (P == 77) return 75557863725914323419136;
88  if (P == 78) return 151115727451828646838272;
89  if (P == 79) return 302231454903657293676544;
90  if (P == 80) return 604462909807314587353088;
91  if (P == 81) return 1208925819614629174706176;
92  if (P == 82) return 2417851639229258349412352;
93  if (P == 83) return 4835703278458516698824704;
94  if (P == 84) return 9671406556917033397649408;
95  if (P == 85) return 19342813113834066795298816;
96  if (P == 86) return 38685626227668133590597632;
97  if (P == 87) return 77371252455336267181195264;
98  if (P == 88) return 154742504910672534362390528;
99  if (P == 89) return 309485009821345068724781056;
100 if (P == 90) return 618970019642690137449562112;
101 if (P == 91) return 1237940039285380274899124224;
102 if (P == 92) return 2475880078570760549798248448;
103 if (P == 93) return 4951760157141521099596496896;
104 if (P == 94) return 9903520314283042199192993792;
105 if (P == 95) return 19807040628566084398385987584;
106 if (P == 96) return 39614081257132168796771975168;
107 if (P == 97) return 79228162514264337593543950336;
108 if (P == 98) return 158456325028528675187087900672;
109 if (P == 99) return 316912650057057350374175801344;
110 if (P == 100) return 633825300114114700748351602688;
111 if (P == 101) return 1267650600228229401496703205376;
112 if (P == 102) return 2535301200456458802993406410752;
113 if (P == 103) return 5070602400912917605986812821504;
114 if (P == 104) return 10141204801825835211973625643008;
115 if (P == 105) return 20282409603651670423947251286016;
116 if (P == 106) return 40564819207303340847894502572032;
117 if (P == 107) return 81129638414606681695789005144064;
118 if (P == 108) return 162259276829213363391578010288128;
119 if (P == 109) return 324518553658426726783156020576256;
120 if (P == 110) return 649037107316853453566312041152512;
121 if (P == 111) return 1298074214633706907132624082305024;
122 if (P == 112) return 2596148429267413814265248164610048;
123 if (P == 113) return 5192296858534827628530496329220096;
124 if (P == 114) return 10384593717069655257060992658440192;
125 if (P == 115) return 20769187434139310514121985316880384;
126 if (P == 116) return 41538374868278621028243970633760768;
127 if (P == 117) return 83076749736557242056487941267521536;
128 if (P == 118) return 166153499473114484112975882535043072;
129 if (P == 119) return 332306998946228968225951765070086144;
130 if (P == 120) return 664613997892457936451903530140172288;
131 if (P == 121) return 1329227995784915872903807060280344576;
132
```

Figure 4. Vertical induction of the six segments in *reductus*.

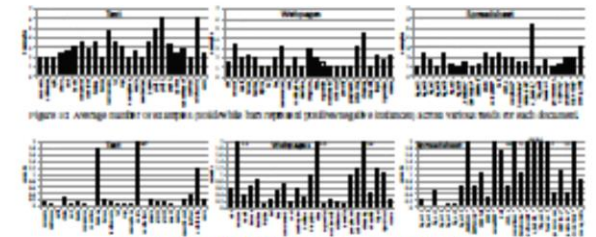


Figure 11: Average learning time in the last interaction across various tasks for each discourse.

Key requirement: each of the three instantiations of *readers* had to successfully able to synthesize a desired text abstraction program for the various tasks in the *single-line* domain. Thus, *readers* tract supports data abstraction tools that are expressive enough to describe a variety of real-world data as traction tasks.

Number of Samples—radar track required an average of 2.6 samples per track across all documents (Fig. 2). Below the average number of samples per track are the results for documents in a given document type. For positive-negative statements (or each of the 10 documents in our dataset), we observed that users have to give more samples to extract data from text files because the structure or layout there is more arbitrary. In contrast, webpages and spreadsheets are generally more structured (because of format tags and the column layout of data) so that users require fewer samples to extract data from documents respectively (i.e., thus requiring fewer samples).

[illegible]

7. Related Work

[illegible]

significantly, primary data techniques are specialized to an underlying context, while ours is more general and can be applied to any context with minimal context-specific changes.

Realized traces in transmission capability actually compresses the transmission capability to a primitive trace. In fact, we have combined them together to provide a better and forward user experience, for example, users using realized trace to extract data never lose the trace, the user can perform using transmission capabilities or number transmission capabilities to modify the extracted trace, and can generate even values in trace editing. By an example, realized trace can be used to highlight some information in the trace, such as the information of the transmission capabilities. These can be used to perform transmission on user objects and these changes are pushed back to the underlying document.

Keynote 4: Highlighting by Examples In *SAINT* and *WE*, authors used to highlight instances or regions in text and *WE* authors using a grammar and an accompanying sampler (in addition to an expected grammar and regular expression models), and *SAINT* authors using the regular expression model. Various authors used *WE* to highlight a set of words in a text and compare the highlighted regions to expected regions. *SAINT* allows users to cover and not highlight instances in text from using much more examples. In contrast, we show the user to provide examples in the original input document and not in the highlighting model. In *WE*, the user is required to provide the user to provide a list of words or phrases or a list of instances of the word that they want to highlight on the document. *WE* also provides instances used to be a strict part of the user or a possible instance. These flexible interactions is enabled by allowing the user to associate

Data Extraction from Log Files: The *python* program [\[23\]](#) has installed automatically on all the data-mining hosts, and it processes

Warning

- Convert-String
- ConvertFrom-String
- ConvertFrom-StringData

Format-Hex

```
PS C:\> "hello world" | Format-Hex
```

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	
00000000	68	65	6C	6C	6F	20	77	6F	72	6C	64						hello world

```
PS C:\> Format-Hex -Path 'C:\Users\slee\Downloads\DSP1009_1 1 1.doc'
```

Path: C:\Users\slee\Downloads\DSP1009_1 1 1.doc

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	
00000000	D0	CF	11	E0	A1	B1	1A	E1	00	00	00	00	00	00	00	00	ÐÏ.à;±.á.....
00000010	00	00	00	00	00	00	00	00	3E	00	03	00	FE	FF	09	00>...þ...
00000020	06	00	00	00	00	00	00	00	00	00	00	00	08	00	00	00

Cryptographic Cmdlets (CMS - RFC5652)

- Get-CmsMessage [-Path] <string>
- Protect-CmsMessage [-To] <CmsMessageRecipient[]>
[-Content] <string> [[-OutFile] <string>]
- Unprotect-CmsMessage [-Content] <string>
[[[-To] <CmsMessageRecipient[]>] [-IncludeContext]

Cryptographic Cmdlets

```
106 [C:\temp]
>> $protected = "Hello World" | Protect-CmsMessage -To "*me@somewhere.com*"

107 [C:\temp]
>> $protected

-----BEGIN CMS-----|
MIIBqAYJKoZIhvcNAQcDoIIBmTCCAZUCAQAxggFQMIIBTAIBADA0MCAxHjAcBgNVBAMMFwx1ZWlv
bG1AbWljcm9zb2Z0LmNvbQIQQYHsbcXnjIJCtH+OhGmc1DANBgkqhkiG9w0BAQcwAASCAQAnkFHM
proJnFy4geFGfyNmxH3yeoPvwEYzdnsoVqqDPAd8D3wao77z70hJEXwz9GeFLnxD6djKV/tF4PxR
E27aduKSLbnxfpf/sepZ4fUkuGibnwWFrXGE3B1G26MCenHWjYQiqv+Nq32Gc97qEAERrhLv6S4R
G+2dJEnesW8A+z9QPo+DwYU5FzD0Td0ExrkswVckpLNR6j17Yaags3ltNVmbdEXekhi6Psf2MLMP
TS0791v2L0KeXFGuPOrdzPAwCkV0vNEqTEBeDnZGrjv/5766bM3GW34FXApod9u+VSFpBnqVOCBA
DVDraA6k+xwBt66cV840HLkh0kT02SIHMDwGCSqGSIb3DQEHAAdBgIghkgBZQMEASoEEJbJaiRl
KMnBoD1dkb/FzSWAEBaL8xkFwCu0e1ZtDj7nSJc=
-----END CMS-----

108 [C:\temp]
>> $protected | Unprotect-CmsMessage
Hello World
```

OData Utils

\$uri =

"http://services.odata.org/V3/(S(readwrite))/OData/OData.svc/"

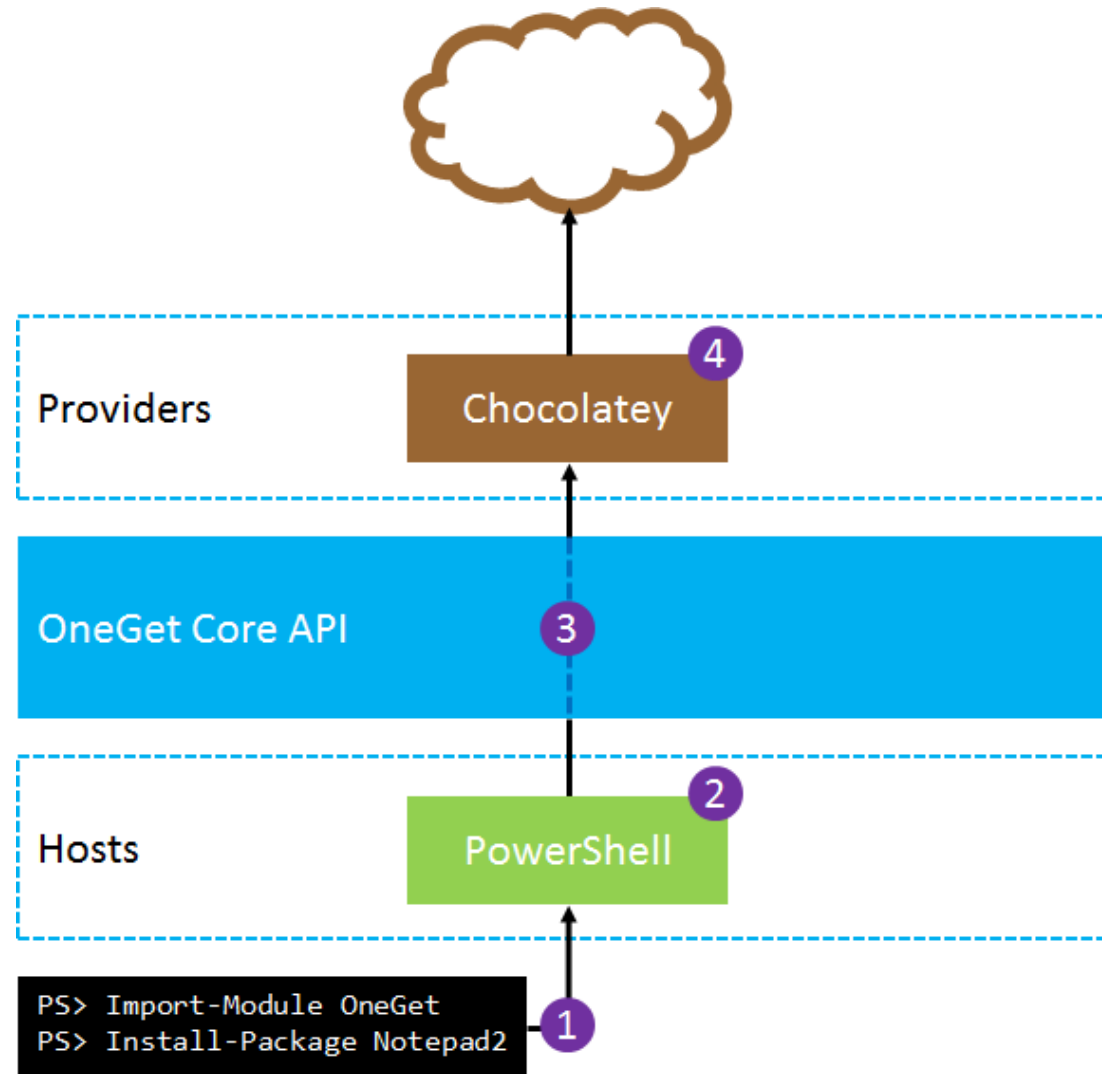
Export-ODataEndpointProxy -Uri \$uri -OutputModule .\Service

DEMO

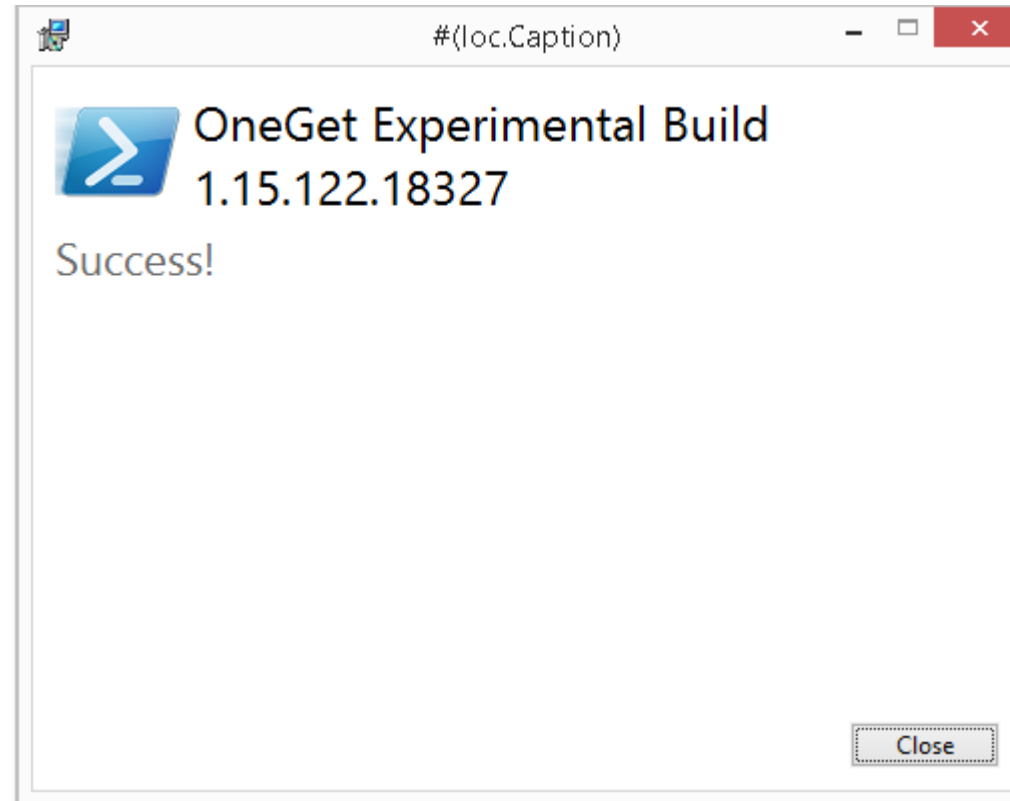
Web Services in PowerShell

- SOAP / XML Web Services
 - New-WebServiceProxy
- OASIS OData
 - Export-ODataEndpointProxy
- Generic REST
 - Invoke-RestMethod
- Generic HTTP
 - Invoke-WebRequest
 - ConvertFrom-Json
 - ConvertTo-Json
 - Select-Xml

OneGet

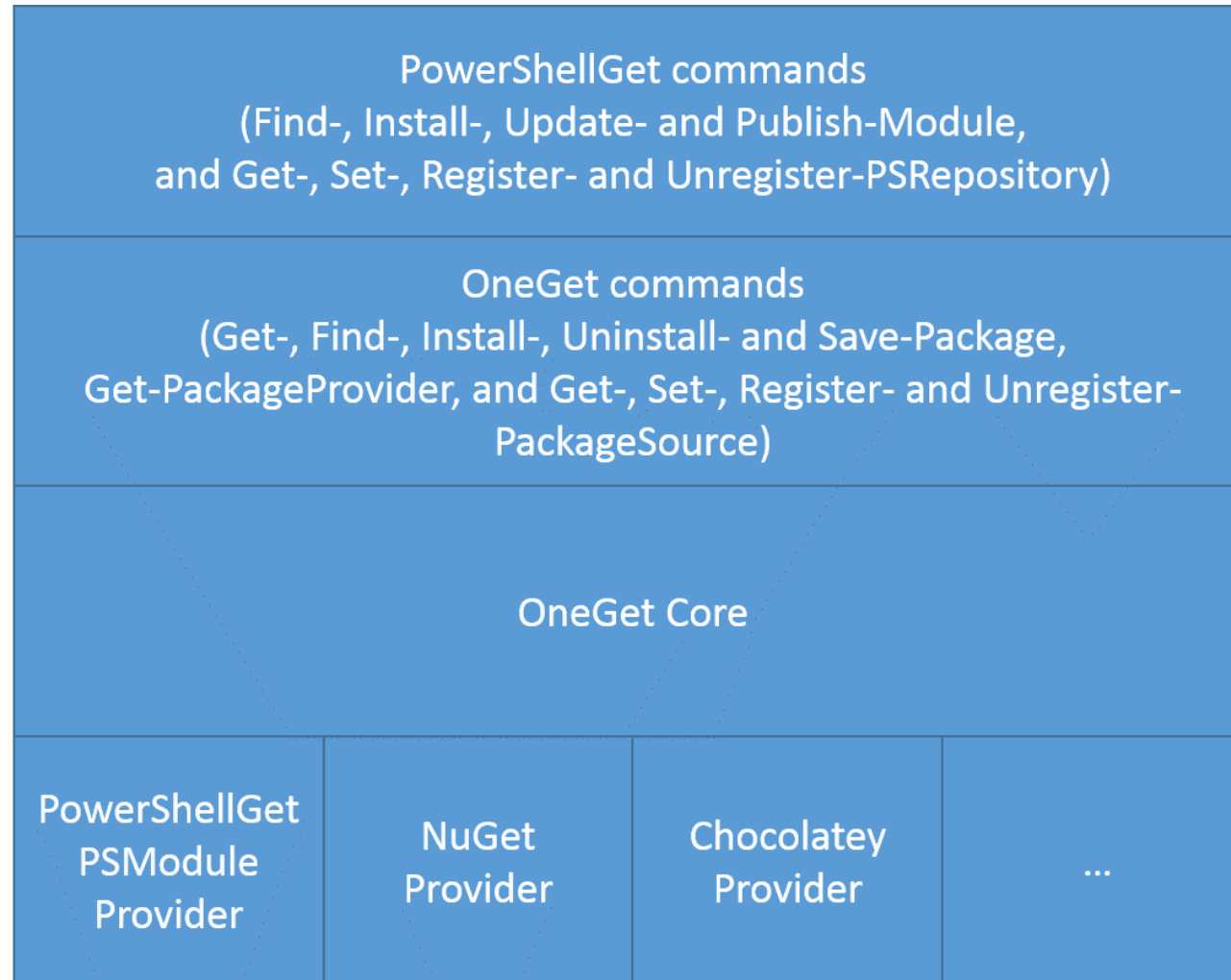


Installing from GitHub



DEMO

PowerShellGet



GOPAS



DAQUAS



Microsoft

PowerShellGet Gallery

This gallery is under limited preview and for demonstration purposes only.

PowerShell Resource Gallery Preview

Poshoholic | Sign out

- Home
- Packages
- Upload Package
- Statistics

What is PowerShell Resource Gallery?

The PowerShell Resource Gallery is the central package repository for sharing and consuming PowerShell Modules, which include PowerShell Desired State Configuration (DSC) Resources.

Using the PowerShell Resource Gallery

Before interacting with the PowerShell Resource Gallery, you must first install the **PowerShellGet** module. **PowerShellGet** comes with the Windows Management Framework (WMF) 5.0 Preview. Once the WMF Preview is installed, you will be able to use the following cmdlets from the PowerShellGet Module:

- Find-Module
- Install-Module
- Publish-Module
- Update-Module

Want to know more about how to use PowerShellGet? [Click here](#) or run `Update-Help -Module PowerShellGet`.

[Install Windows Management Framework 5.0 Preview](#)

Unique Packages	Total Package Downloads	Total Packages
88	8,351	166

SxS Module Version

```
PS C:\windows\system32> Get-Module -ListAvailable -Name ContosoServer | Format-List  
Name,Version,ModuleBase
```

```
Name      : ContosoServer
```

```
Version    : 2.0
```

```
ModuleBase : C:\Program Files\WindowsPowerShell\Modules\ContosoServer\2.0
```

```
Name      : ContosoServer
```

```
Version    : 1.0
```

```
ModuleBase : C:\Program Files\WindowsPowerShell\Modules\ContosoServer\1.0
```

Switch Management

```
PS> Get-Command *-NetworkSwitch*
```

CommandType	Name	Source
-----	----	-----
Function	Disable-NetworkSwitchEthernetPort	NetworkSwitch
Function	Disable-NetworkSwitchFeature	NetworkSwitch
Function	Disable-NetworkSwitchVlan	NetworkSwitch
Function	Enable-NetworkSwitchEthernetPort	NetworkSwitch
Function	Enable-NetworkSwitchFeature	NetworkSwitch
Function	Enable-NetworkSwitchVlan	NetworkSwitch
Function	Get-NetworkSwitchEthernetPort	NetworkSwitch
Function	Get-NetworkSwitchFeature	NetworkSwitch
Function	Get-NetworkSwitchGlobalData	NetworkSwitch
Function	Get-NetworkSwitchVlan	NetworkSwitch
Function	New-NetworkSwitchVlan	NetworkSwitch
Function	Remove-NetworkSwitchEthernetPortIPAddress	NetworkSwitch
Function	Remove-NetworkSwitchVlan	NetworkSwitch
Function	Restore-NetworkSwitchConfiguration	NetworkSwitch
Function	Save-NetworkSwitchConfiguration	NetworkSwitch
Function	Set-NetworkSwitchEthernetPortIPAddress	NetworkSwitch
Function	Set-NetworkSwitchPortMode	NetworkSwitch
Function	Set-NetworkSwitchPortProperty	NetworkSwitch
Function	Set-NetworkSwitchVlanProperty	NetworkSwitch

Switch Management

```
PS> $ip = "10.0.0.2"
PS> $sessionOption = New-CimSessionOption -UseSsl -SkipCACheck -SkipCNCheck -SkipRevocationCheck
PS> $s = New-CimSession -CN $ip -port 5986 -Auth Basic -Credential admin -SessionOption
$sessionOption
```

```
PS> Get-NetworkSwitchFeature -CimSession $s
```

ElementName	InstanceID	FeatureName	IsEnabled	PSComputerName
-----	-----	-----	-----	-----
SSH	Arista:Feature:2	2	True	10.0.0.2
Tacacs	Arista:Feature:3	3	True	10.0.0.2
BGP	Arista:Feature:4	4	False	10.0.0.2
VLAN	Arista:Feature:5	5	True	10.0.0.2
LACP	Arista:Feature:6	6	True	10.0.0.2
DHCP	Arista:Feature:7	7	False	10.0.0.2
LLDP	Arista:Feature:8	8	True	10.0.0.2

Switch Management

```
PS> Get-NetworkSwitchEthernetPort -CimSession $s | Format-Table InstanceID, ElementName, MaxSpeed, PortNumber, EnabledState
```

InstanceID	ElementName	MaxSpeed	PortNumber	EnabledState
-----	-----	-----	-----	-----
Arista:Ether...	Ethernet1	1410065408	1	2
Arista:Ether...	Ethernet2	1410065408	2	2
Arista:Ether...	Ethernet3	1410065408	3	2
Arista:Ether...	Management1	1410065408	97	2

```
PS> Disable-NetworkSwitchEthernetPort -PortNumber 3 -CimSession $s
```

ReturnValue	Job	PSComputerName
-----	---	-----
0		10.0.0.2

Custom Enums and Classes

```
enum Color
{
    Green
    Red
    Blue
}
```

```
class Dog : Pet
{
    Dog([string] $Name) : base($Name)
    {
    }

    [string] Greet()
    {
        return 'Bark'
    }
}
```


DEMO



Gold partner:



Generální partner:



Novinky PowerShell 5 Preview

Michael Grafnetter
www.dsinternals.com

18. – 21. května 2015

Tech·Ed
DevCon 