PowerShell Classes – Onhands with the example LogFileParser

David das Neves
Premier Field Engineer

CHEF™ CHEF.IO

Microsoft

KEMP APPLICATION DELIVERY

SAPIEN Technologies, Inc.

David das Neves
Premier Field Engineer

Twitter: @david_das_neves

LinkedIn: DaviddasNeves

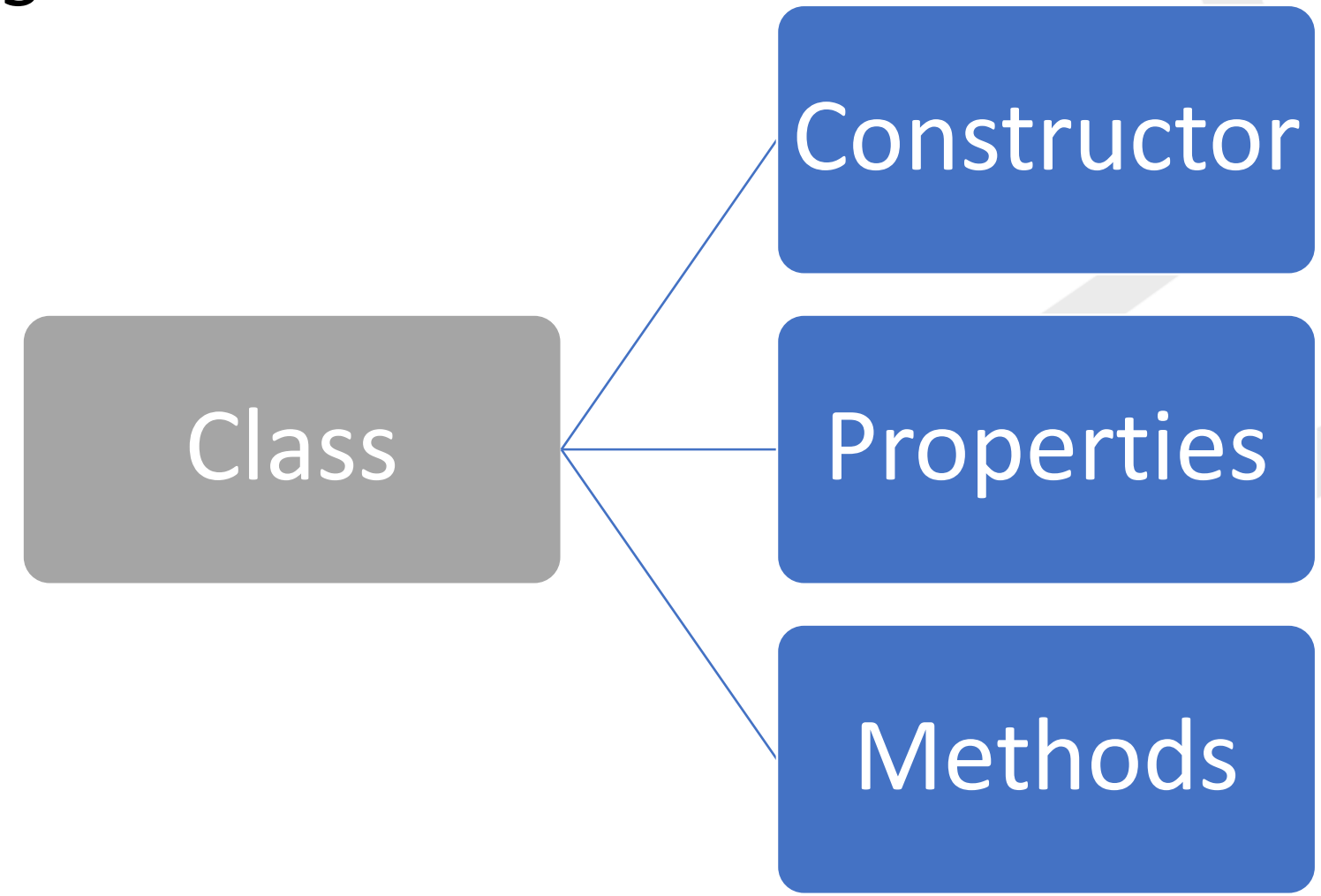
Blog: http://aka.ms/ddnblog

E-Mail: David.dasNeves@Microsoft.com

Introduction

101 PowerShell Classes

Classes



Classes

```
#Class for a single LogFileTypeClass.
class LogFileTypeClass
{
}

[LogFileTypeClass]::new()

[LogFileTypeClass]::new("Parameter")

New-Object -TypeName ,LogFileTypeClass‘ -ArgumentList
$file.FullName, $fileType, $regExString
```

Classes & Instances

A class is a recipe

You can create many dishes out of this

An instance is a dish

you can interact with a dish – like eat it.

Keyword Static

No need to instanciate the object

helper classes

Classes - Constructor

```
#Class for a single LogFileTypeClass.
class LogFileTypeClass
{
  #Constructor without values
  LogFileTypeClass()
  {}
}
```

Classes

```
#Class for a single LogFileTypeClass.
class LogFileTypeClass
{
  #Constructor without values
  LogFileTypeClass()
  {}

  #Constructor with values
  LogFileTypeClass($logFileType, $description, $regExString, $logFiles, $locationsLogFiles)
  {
    $this.LogFileType = $logFileType
    $this.Description = $description
    $this.RegExString = $regExString
    $this.LogFiles = $logFiles
    $this.LocationsLogFiles = $locationsLogFiles
  }
}
```

Properties

```
# LogfileType
[string]$LogFileType

# internal Description of the logFile
hidden [string]$internalDescription

# internal Description of the logFile
static [string]$internalDescription
```

Hidden is not the same as **private** in languages such as C#.
Hidden properties are masked, but still accessible to the debugger.

Methods

```
#Overriding ToString to show the LogFileTypes in the
overview
[string] ToString()
{
   return ($this.LogFileType).ToString()
}

hidden init()
{
   {…}
}
```

Methods - overloaded

```
## standard constructor
 ## LogFileType SCCM is set
 ParsedLogFile($LogFilePath, $regExString)
 {
   $this.LogFileType = 'SCCM'; $this.LogFilePath = $LogFilePath;
   $this.LogFileType = $this.LogFileType; $this.RegExString = $regExString;
   $this.init()
 }

 ## Constructor with LogFileType
 ParsedLogFile($LogFilePath, $logFileType, $regExString)
 {
   $this.LogFilePath = $LogFilePath; $this.LogFileType = $logFileType; $this.RegExString = $regExString
   $this.init()
 }
```

Classes – inherited

```
#Inherited class to add functions to the ParsedLogData
class WindowsUpdateLog : ParsedLogFile
{

}
```

Classes – inherited

```
#Inherited class to add functions to the ParsedLogData
class WindowsUpdateLog : ParsedLogFile
{

  ## LogFileType SCCM is set
  WindowsUpdateLog($LogFilePath, $regExString): base($LogFilePath, $regExString)
  {}

  ## Constructor with LogFileType
  WindowsUpdateLog($LogFilePath, $logFileType, $regExString): base($LogFilePath,
$logFileType, $regExString)
  {}
}
```

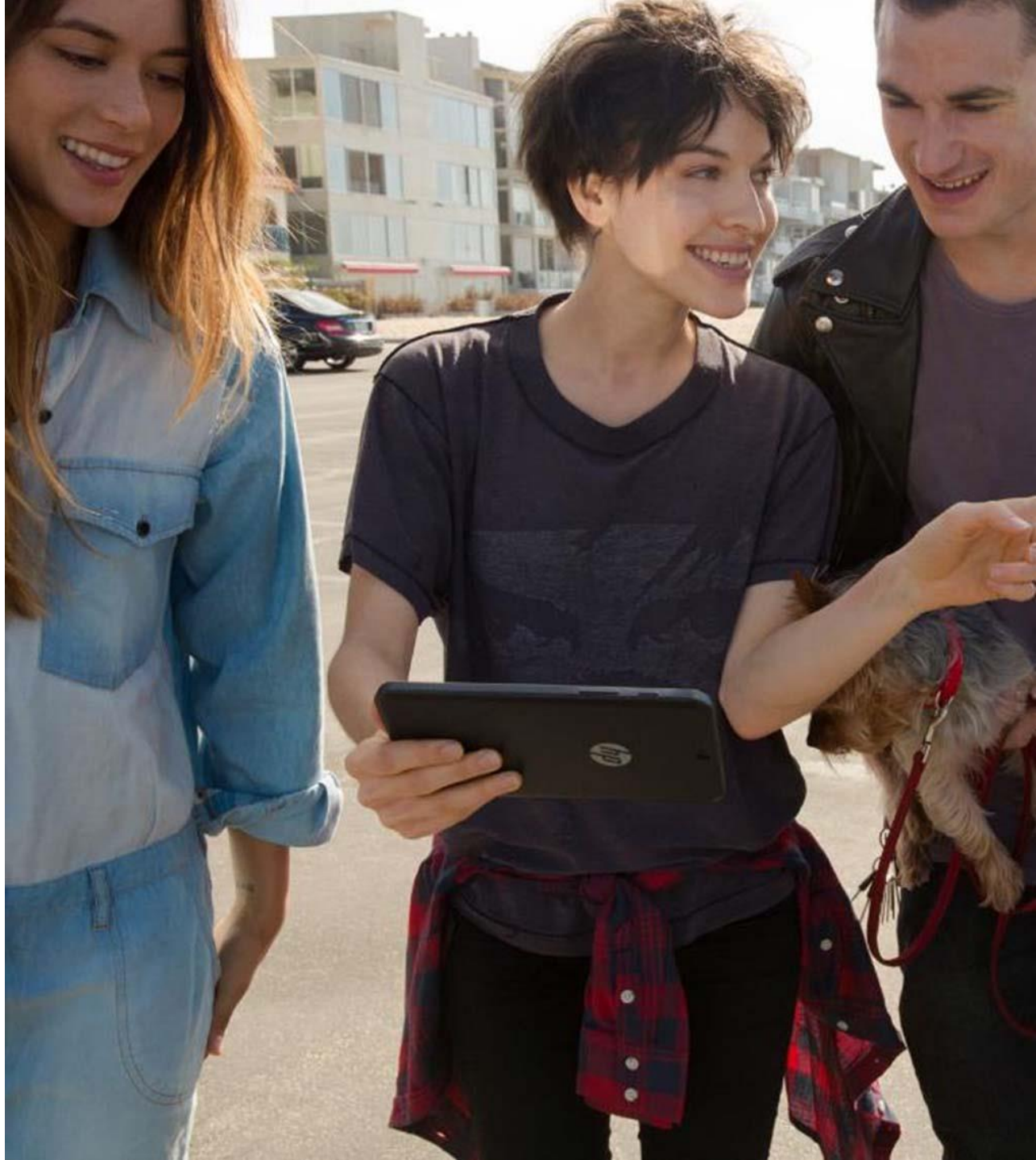
Classes – inherited – additional methods

```
#Inherited class to add functions to the ParsedLogData
class WindowsUpdateLog : ParsedLogFile
{
{…} Constructors

#region Special Public Functions

  #WindowsUpdateLog

  [PSCustomObject]GetInstalledUpdates()
  {
  }
  #endregion
}
```

Classes – inherited – override

```
#Inherited class to add functions to the ParsedLogData
class WindowsUpdateLog : ParsedLogFile
{
   ## Override
   GetParsedLogFile()
   {
      ## something other code
   }
}
```

Demo

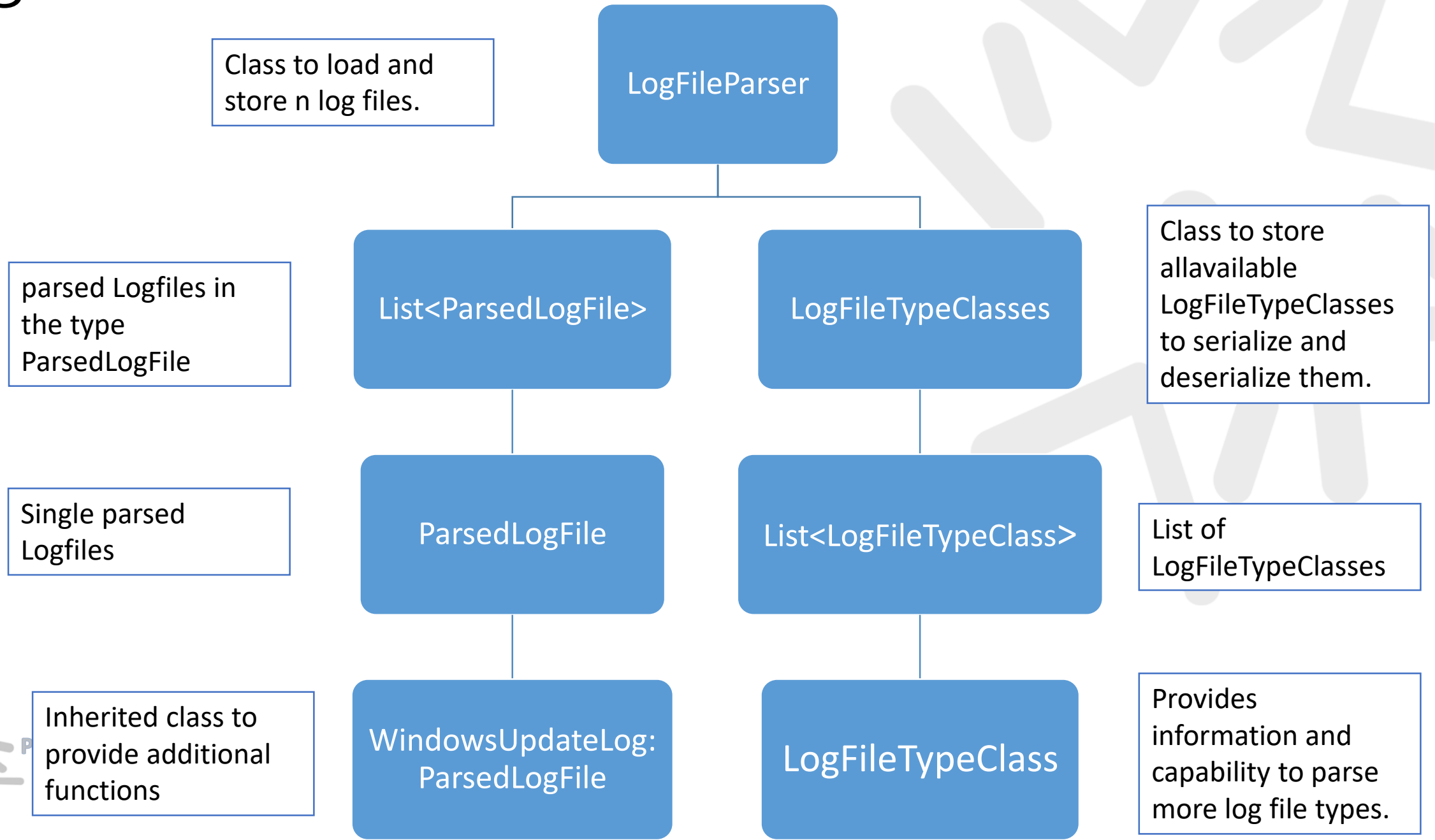


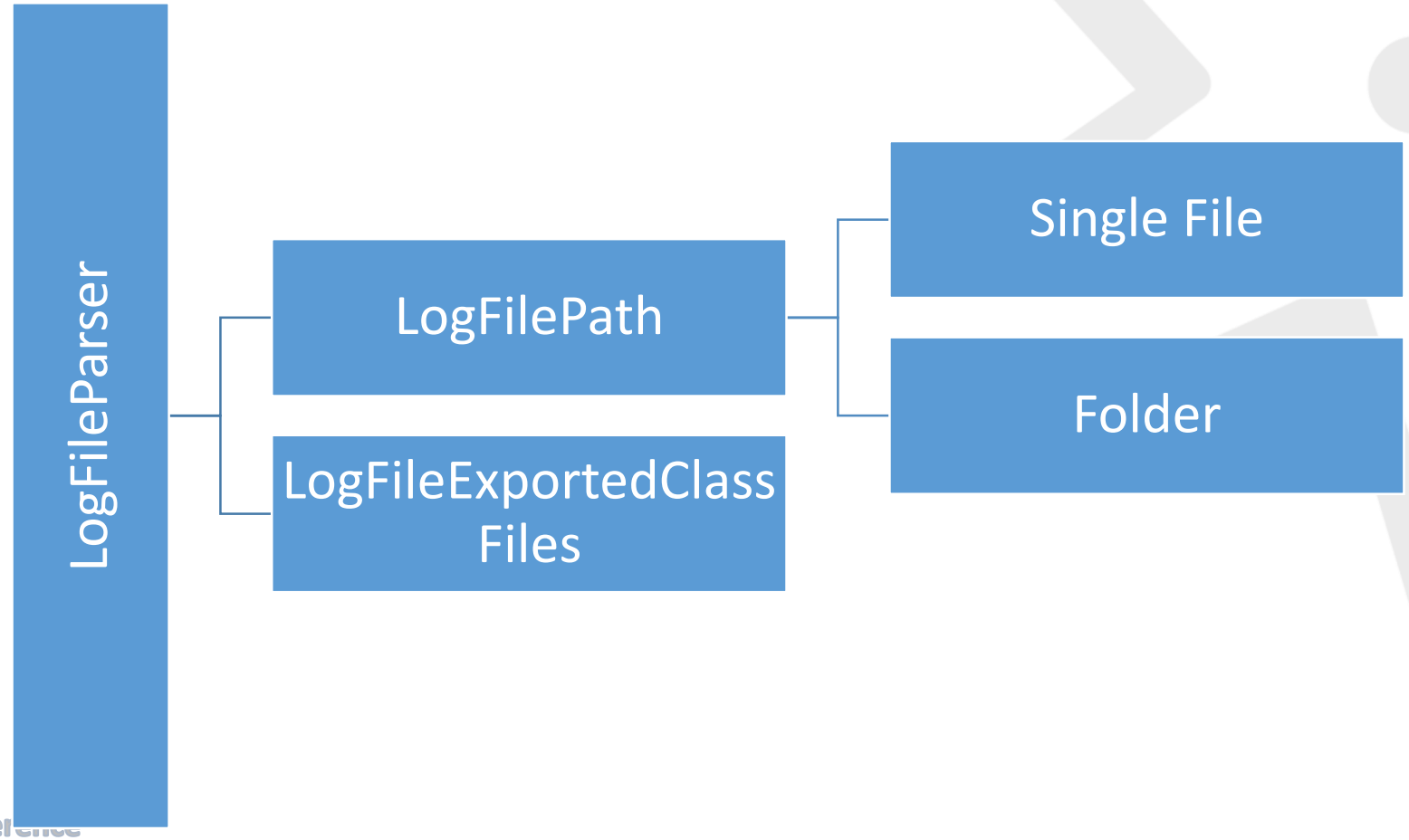
LogFileParser

Why PowerShell Classes?
Why LogFileParser?

- Proof of Concept
- Example with value
- Showing some more complex techniques and their benefits
- Finds usage in DSC → DSC-Resource

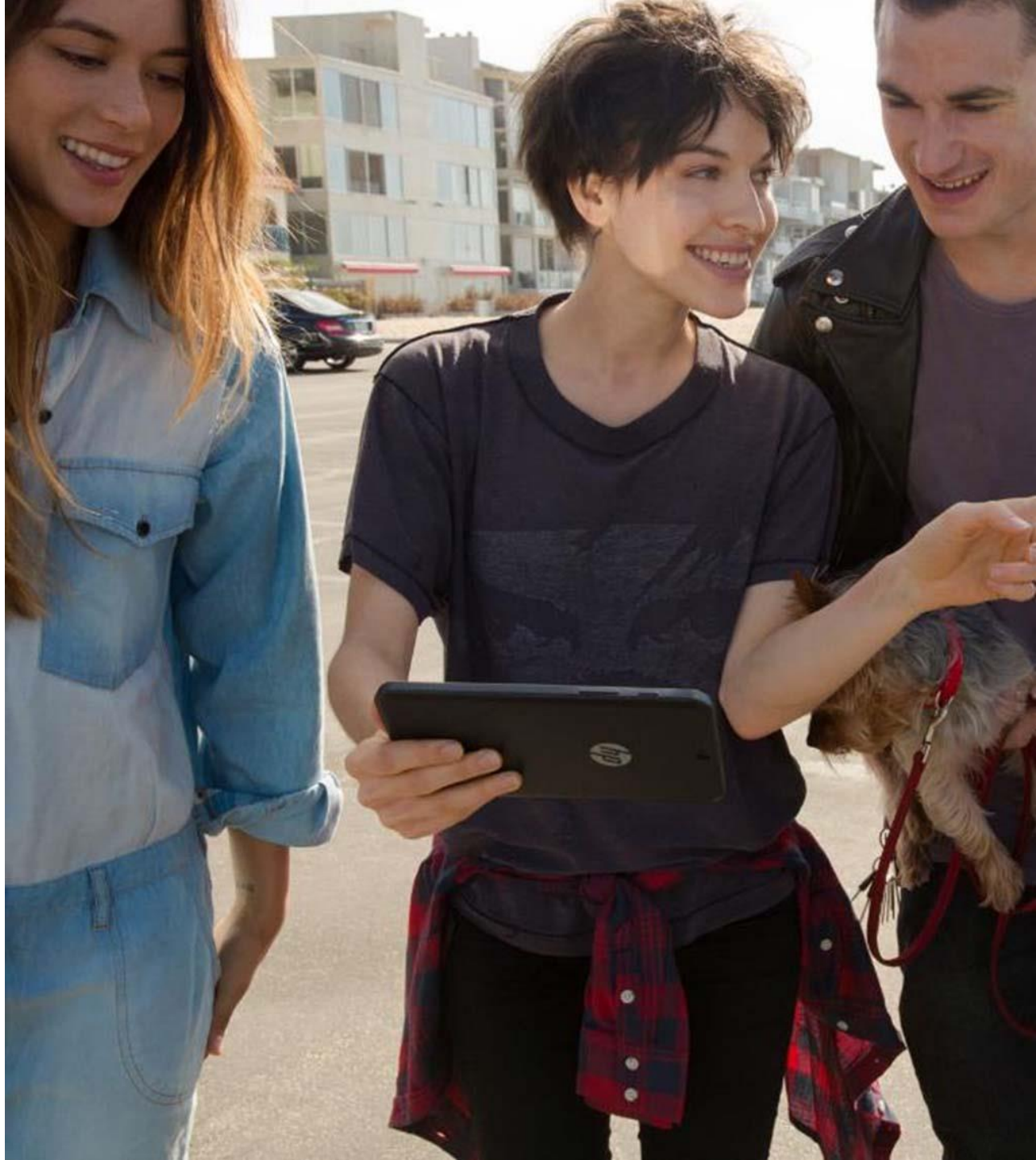
LogFileParser Structure




LogFileParser
LogFilePath
LogFileExportedClass Files
Single File
Folder

Demo

https://blogs.msdn.microsoft.com/daviddasneves/2017/10/27/logfileparser-with-powershell/



Don’t Forget!

- Fill in your survey in Mobile app – it’s how we do better!
- Don’t lose you badge! You need it for the Social Events
- Grab the Speakers for a chat – they all have time for you!
- Let everyone know what they are missing on Social Media

  #PowerShell

  #PSConfAsia

Photos of Marina Bay Credit: Sebastian Szumigalski