Betriebssystem

Inhaltsverzeichnis

- 1 Allgemeine Attribute
- 2 GetEnvironmentVariable
- 3 SetEnvironmentVariable
- 4 GetFingerprint
- 5 ShellExecute
- 6 IsProcessRunning
- 7 IsServiceInstalled
- 8 IsServiceRunning
- 9 ServiceInstall
- 10 ServiceUninstall
- 11 ServiceStart
- 12 ServiceStop
- 13 KillTask
- 14 RegAsm
- 15 RegSvr32
- 16 GetWMIClassInstance
- 17 GetWMIClassInstanceCount
- 18 GetWMIDataValue
- 19 WindowsProductKey
- 20 Parameter

Die Aktionen und Parameter aus der Kategorie Betriebssystem

1 Allgemeine Attribute

Die allgemeinen Attribute **IgnoreError**, **Variable** und **Condition** können bei allen Aktionen angegeben werden. Die Attribute sind optional und brauchen nur bei Bedarf hinterlegt werden. Wenn diese für eine Aktion nicht benötigt werden, können diese aber auch zur besseren Lesbarkeit des Skriptes entfernt werden.

IgnoreError

Das optionale Attribut **IgnoreError** gibt an, ob bei einem Fehler die Ausführung des Batchpad Skriptes abbricht oder das Skript weiter ausgeführt werden soll. Der Wert muss dem Typ Boolean (true oder false) entsprechen.

Variable

Das optionale Attribut **Variable** kann immer dann verwendet werden, wenn man das Ergebnis einer auszuführenden Aktion ermitteln möchte. Variable="{@ResultFileExists}".
Die Ergebnisse sind je nach ausgeführter Aktion vom Typ her unterschiedlich, oft ist es ein Boolean (true oder false) der angibt ob die Aktion erfolgreich war. Bei Aktionen für Zeichenketten sind die Ergebnisse dann eher vom Typ String usw.

Condition

Das optionale Attribut **Condition** gibt an, ob die Aktion ausgeführt werden soll. Hierzu wird der Inhalt des Attributes als logischer Ausdruck auf Wahr oder Falsch geprüft. Der Ausdruck sollte dem Typ Boolean (true oder false) entsprechen.
Der Ausdruck kann Funktionen aus VBScript enthalten, genauso wie Operatoren NOT, OR, AND...
Mit dem Condition Attribut wertet man in der Regel Variablen aus, die Ergebnisse aus zuvor durchgeführten Aktionen enthalten. Beispiel: Condition="NOT {@ResultFileExists}"

2 GetEnvironmentVariable

```
<GetEnvironmentVariable Name="" TargetProcess="false" TargetUser="false" TargetSystem="false" Condition="" Variable="{@Result}" IgnoreError="false" />
```

3 SetEnvironmentVariable

```
<SetEnvironmentVariable Name="" Value="" TargetProcess="false" TargetUser="false" TargetSystem="false" Condition="" Variable="{@Result}" IgnoreError="false" />
```

4 GetFingerprint

<GetFingerprint Key="" Condition="" Variable="{@Result}" IgnoreError="false" />

5 ShellExecute

<ShellExecute File="" Arguments="" OutputData="{@OutputData}" ErrorData="{@ErrorData}" CreateNoWindow="true" RedirectStandardOutput="true" RedirectStandardError="true" UseShellExecute ="false" Condition="" Variable="{@Result}" IgnoreError="false" />

6 IsProcessRunning

<IsProcessRunning ProcessName="" Condition="" Variable="{@Result}" IgnoreError="false" />

7 IsServiceInstalled

<IsServiceInstalled ServiceName="" Condition="" Variable="{@Result}" IgnoreError="false" />

8 IsServiceRunning

<IsServiceRunning ServiceName="" Condition="" Variable="{@Result}" IgnoreError="false" />

9 ServiceInstall

<ServiceInstall ServicePath="" ServiceFile="" ServiceName="" Condition="" Variable="{@Result}" IgnoreError="false" />

10 ServiceUninstall

<ServiceUninstall ServicePath="" ServiceFile="" ServiceName="" Condition="" Variable="{@Result}" IgnoreError="false" />

11 ServiceStart

```
<ServiceStart ServiceName="" WaitForStatusRunning="false" Condition="" Variable="{@Result}" IgnoreError
="false" />
```

12 ServiceStop

```
<ServiceStop ServiceName="" WaitForStatusStopped="false" Condition="" Variable="{@Result}" IgnoreError
="false" />
```

13 KillTask

```
<KillTask Process="" Condition="" Variable="{@Result}" IgnoreError="false" />
```

14 RegAsm

```
<RegAsm File="" Arguments="" OutputData="{@OutputData}" ErrorData="{@ErrorData}" Condition=""
Variable="{@Result}" IgnoreError="false" />
```

15 RegSvr32

```
<RegSvr32 File="" Arguments="" OutputData="{@OutputData}" ErrorData="{@ErrorData}" Condition=""
Variable="{@Result}" IgnoreError="false" />
```

16 GetWMIClassInstance

```
<GetWMIClassInstance Data="{@myList}" Class="Win32_Processor" Instance="0" Variable="{@Result}" />
```

17 GetWMIClassInstanceCount

```
<GetWMIClassInstanceCount Class="Win32_Processor" Variable="{@Result}" />
```

18 GetWMIDataValue

```
<GetWMIDataValue Data="{@myList}" Key="" Variable="{@Result}" />
```

19 WindowsProductKey

```
<WindowsProductKey Variable="{@Result}" />
```

20 Parameter

Parameter	Wert (Beispiel)
{@CommandLine}	"C:\Program Files (x86)\LogiSoft Batchpad\Batchpad.exe"
{@ComputerName}	DEVWIN10
{@ComputerUser}	LogiSoft
{@ComputerDomain}	DEVWIN10
{@SystemName}	Microsoft Windows 10 Pro
{@SystemVersion}	6.2.9200.0
{@Is64BitOperatingSystem}	True
{@InstalledUICulture}	de-DE