

Windows: Der Aufruf der Verwaltung in der Systemsteuerung funktioniert nicht mehr und erzeugt einen Fehler

geschrieben von Andreas Potthoff | 30. Juli 2017

Wer es beim Aufräumen von Windows etwas zu gut gemeint hat kann Probleme bekommen. Deshalb sollte man bei der Verwendung von sogenannten Optimierungs-Tools sehr vorsichtig sein. Man kann sich damit auch das ganze System zerschießen, insbesondere wenn an der Registry herumoptimiert wird.

Hier eine Möglichkeit folgenden Fehler wieder loszuwerden:

Fehlermeldung:

Der Pfad ist nicht verfügbar.

C:\ProgramData\Microsoft\Windows\Start Menu\Programs\Administrative Tools bezieht sich auf einen Pfad, der nicht verfügbar ist. Dieser kann auf einer Festplatte dieses Computers oder im Netzwerk sein. Stellen Sie sicher, dass der Datenträger richtig eingelegt ist, bzw. dass eine Verbindung mit dem Internet oder dem eigenen Netzwerk besteht und wiederholen Sie den Vorgang. Es kann sein, dass die Informationen in einen anderen Pfad verschoben wurden, wenn der Pfad weiterhin nicht ermittelt werden kann.

Der Ordner *Verwaltung* ist kein gewöhnlicher Ordner, sondern ein Systemordner auf den aus der Systemsteuerung zugegriffen wird. Löscht man diesen oder werden entsprechende Registryeinträge geändert, erscheint beim Öffnen der *Verwaltung* der obige Fehler.

Lösung:

Das Systemdatei-Überprüfungsprogramms (sfc.exe) wird zur Problembehandlung bei fehlenden oder beschädigten Systemdateien verwendet und ist unter Windows Vista, 7, 8, 8.1, 10 als Bordmittel zu finden.

Man startet ein Kommandozeilenfenster mit Eingabeaufforderung (Administrator) und gibt folgenden Befehl ein:

```
sfc /scannow
```

Jetzt wird das System gescannt und versucht Fehler zu beheben. Wenn nicht alle Fehler repariert werden können, sollte man einen Neustart des Systems durchführen und ggf. den Prozess mehrmals wiederholen. An den Programmen, Dateien und persönlichen Einstellungen wird nichts verändert – es werden nur Systemdateien repariert.

Hinweis: Schließen Sie das Eingabeaufforderungsfenster erst, wenn die Überprüfung zu 100 % abgeschlossen ist. Die Überprüfungsergebnisse werden angezeigt, sobald dieser Prozess abgeschlossen ist.

Pflege:

Manchmal bemerkt man kleine Fehler gar nicht, daher ist ein regelmäßiger Scan (z.B. wöchentlich) vielleicht gar nicht so unsinnig, da Kleinigkeiten sofort repariert werden. Somit können auch Fehleransammlungen vermieden werden.

Windows: Systempflege – Mehr Festplattenspeicher durch das

Löschen alter NVIDIA Software und Treiber

geschrieben von Andreas Potthoff | 30. Juli 2017

Jedes mal wenn ein neuer Geforce-Treiber installiert wird bleiben die alten Dateien auf Ihrem Windows-System zurück. Dieses Problem besteht schon seit Jahren. Mit der Einführung von NVIDIA Geforce Experience gibt es noch mehr unnötigen Datenmüll.

Durch das Löschen der alten NVIDIA Treiberdateien lässt sich viel Speicherplatz sparen – hier können schon einige Gigabyte (GB) zusammenkommen, insbesondere wenn man die automatische Update-Funktion von Geforce Experience verwendet.

Hier findet ihr einen kleinen Leitfaden um den Datenballast loszuwerden und Festplattenspeicher freizugeben.

Bitte hinterlasst einen Kommentar wie viel GB euch die Säuberungsaktion der Festplatte gebracht hat. Danke!

Update: Ab GeForce Experience 3.9.0 hat NVIDIA ein Bereinigungstool hinzugefügt, welches alte Treiberversionen automatisch entfernt. Die NVIDIA-Software behält jetzt nur Installer für die aktuelle und vorherige Version des Treibers bei, die insgesamt etwa 1 GB betragen.

Festplattenspeicher freigeben:

1. Temporäres Installationsverzeichnis

C:\NVIDIA

Sie können das Verzeichnis und die darin enthaltenen Dateien und Unterverzeichnisse löschen, da dieses Verzeichnis nur für die Installation des Grafikkartentreibers benötigt wird.

2. Geforce Experience

C:\Programme\NVIDIA Corporation\Installer2

Löschen sie alle Dateien und Verzeichnisse innerhalb dieses Ordners – aber nicht den Ordner selbst!

3. Geforce Experience Download Verzeichnis

C:\ProgramData\NVIDIA Corporation\NetService

Dieser Ordner ist ein versteckter Ordner, der nur mit entsprechenden Einstellungen im Explorer angezeigt wird. In diesem Ordner können alle *.exe Dateien gelöscht werden.

Mit der GFE 3.x Version wurde dieser Ordner geändert.

C:\ProgramData\NVIDIA Corporation\Downloader

Die meisten Unterverzeichnisse beinhalten ausführbare Installationsdateien. Löschen Sie alle Unterverzeichnisse mit zufälligen Namen (z.B. *0dfe1295f805b273217a4eb88f7cf66d*) außer die Unterverzeichnisse *config* und *latest*.

DMOZ – Das Open Directory Project (Webverzeichnis) wurde geschlossen

geschrieben von Andreas Potthoff | 30. Juli 2017

DMOZ (Directory Mozilla) war das größte von Menschen gepflegte Webverzeichnis des World Wide Web. Die Inhalte des Open Directory Projekts (ODP) waren frei und wurden von ehrenamtlichen Editoren bearbeitet und aktualisiert. Am 28. Februar 2017 wurde bekannt, dass das Verzeichnis zum 14. März 2017 geschlossen werde. Über die Gründe der Schließung des Webverzeichnis wurde von offizieller Seite nichts

bekanntgegeben.

Geschichte:

Das ODP wurde am 5. Juni 1998 online gestellt und ursprünglich unter dem Namen Gnuhoo gegründet. Der Name wurde später aus rechtlichen Gründen zu Newhoo geändert. Newhoo war ein starker Konkurrent zu dem Yahoo Webverzeichnis – daher auch der Name. Im Oktober 1998 wurde Newhoo von Netscape gekauft und hieß dann Open Directory Project. Nur einen Monat später wurde Netscape von AOL gekauft, welches später mit Time Warner fusionierte. In diesem Jahr startete auch der Hauptkonkurrent – Google. Die Hostingadresse dmoz.org wurde von directory.mozilla.org abgeleitet.

Einige Wahrscheinlichkeiten:

Das ODP konnte nicht alle Anfragen erfüllen, wurde selten aktualisiert, war schlecht gepflegt und war mit Editoren unterbesetzt. Korrupte Editoren haben Webseiten mit geringem Mehrwert für die Öffentlichkeit eingetragen. Dadurch war das ODP für seine Benutzer nicht mehr wirklich nützlich was zum Untergang des Projekts geführt hat.

Links:

Einige ehemalige Redakteure von DMOZ bieten eine statische Kopie (Spiegelseite) oder eine Weiterführung vom DMOZ.org Webverzeichnis an:

- <http://dmaz-odp.org/>
 - <http://www.opendirectoryproject.org/>
 - <https://curlie.org/>
 - <http://dmaztools.net/>
-

PHP: Funktion: Aus einer TXT-Datei eine HTML-Tabelle erstellen

geschrieben von Andreas Potthoff | 30. Juli 2017

Für mein **Pi Fusion** Projekt habe ich folgende Funktion in Raspcontrol gefunden, die aus einer Textdatei (bzw. aus einem String oder einer Standardausgabe bei Linux) eine HTML-Tabelle macht. Ich habe diese Funktion für meine Bedürfnisse leicht modifiziert. Die Ausgabe funktioniert gut, habe aber diese Funktion ausgelagert, da ich diese Funktion bereits durch eine andere ersetzt habe. Die TXT-Datei (\$txtFile) muss eine Spaltenüberschrift haben, damit die Funktion auch korrekt arbeitet.

```
// function by raspcontrol - modified by Andy_P
function txt_to_html_table($txtFile)
{
    $txtFile = preg_split('/[\r\n]+/', $txtFile);

    // remove double (or more) spaces for all items
    foreach ($txtFile as &$item) {
        $item = preg_replace('/[[:blank:]]+/', ' ', $item);
        $item = trim($item);
    }

    // remove empty lines
    $txtFile = array_filter($txtFile);

    // the first line contains titles
    $columnCount = preg_match_all('/\s+/', $txtFile[0]);
    $txtFile[0] = '<tr><th>' . preg_replace('/\s+/',
'</th><th>', $txtFile[0], $columnCount) . '</th></tr>';
    $tableHead = $txtFile[0];
    unset($txtFile[0]);

    // others lines contains table lines
```

```

foreach ($txtFile as &$item) {
    $item = '<tr><td>' . preg_replace('/\s+/', '</td><td>',
$item, $columnCount) . '</td></tr>';
}

// return the build table
return '<table class=\'table table-striped\'>'
    . '<thead>' . $tableHead . '</thead>'
    . '<tbody>' . implode($txtFile) . '</tbody>'
    . '</table>';
}

```

Beispiel:

```

$storageDetails = shell_exec('df -lTh');
$result = txt_to_html_table($storageDetails);

```

So sieht dann die Ausgabe der Tabelle nach einigen kosmetischen Veränderungen aus:

Storage usage by partitions						
Filesystem	Type	Size	Used	Avail	Use%	Mounted
/dev/root	ext4	28G	4.1G	23G	16%	/
devtmpfs	devtmpfs	458M	0	458M	0%	/dev
tmpfs	tmpfs	462M	0	462M	0%	/dev/shm
tmpfs	tmpfs	462M	6.3M	456M	2%	/run
tmpfs	tmpfs	5.0M	4.0K	5.0M	1%	/run/lock
tmpfs	tmpfs	462M	0	462M	0%	/sys/fs/cgroup
/dev/mmcblk0p6	vfat	65M	21M	45M	33%	/boot
tmpfs	tmpfs	93M	0	93M	0%	/run/user/1000

Credit: Raspcontrol, Jacob Clark (Bioshox), GNU GPL 2.0

Wie kann ich einen Page

Reload mit JavaScript machen?

geschrieben von Andreas Potthoff | 30. Juli 2017

Hier einige Beispiele wie man mit JavaScript eine Seite reloaden kann:

Links:

```
<a href="javascript:location.reload(true)">Reload Page</a>
```

```
<a href="javascript:refresh()">Reload Page</a>
```

Buttons:

```
<input type="button" value="Reload Page"
onClick="window.location.reload()">
```

```
<input type="button" value="Reload Page"
onClick="history.go(0)">
```

```
<input type="button" value="Reload Page"
onClick="window.location.href=window.location.href">
```

```
<input type="button" value="Reload Page"
onClick="document.location.reload(true)">
```

```
<input type="button" value="Reload Page" onclick="refresh()">
```

Die Beispiele können auch entsprechend angepasst werden. Hier noch 535 weitere Möglichkeiten:

```
location = location
location = location.href
location = location
location = self.location
location = self.location.href
location = self.location
location = self
location = self.href
location = self
location = window.location
```



```
location = window.location.href
location = window.location
location = window
location = window.href
location = window
location.assign(location)
location.assign(location.href)
location.assign(location)
location.assign(self.location)
location.assign(self.location.href)
location.assign(self.location)
location.assign(self)
location.assign(self.href)
location.assign(self)
location.assign(window.location)
location.assign(window.location.href)
location.assign(window.location)
location.assign(window)
location.assign(window.href)
location.assign(window)
location.href = location
location.href = location.href
location.href = location
location.href = self.location
location.href = self.location.href
location.href = self.location
location.href = self
location.href = self.href
location.href = self
location.href = window.location
location.href = window.location.href
location.href = window.location
location.href = window
location.href = window.href
location.href = window
location.reload()
location.replace(location)
location.replace(location.href)
location.replace(location)
location.replace(self.location)
location.replace(self.location.href)
```

```
location.replace(self.location)
location.replace(self)
location.replace(self.href)
location.replace(self)
location.replace(window.location)
location.replace(window.location.href)
location.replace(window.location)
location.replace(window)
location.replace(window.href)
location.replace(window)
location(location)
location(location.href)
location(location)
location(self.location)
location(self.location.href)
location(self.location)
location(self)
location(self.href)
location(self)
location(window.location)
location(window.location.href)
location(window.location)
location(window)
location(window.href)
location(window)
location = location
location = location.href
location = location
location = self.location
location = self.location.href
location = self.location
location = self
location = self.href
location = self
location = window.location
location = window.location.href
location = window.location
location = window
location = window.href
location = window
location()
```

```
location(location)
location(location.href)
location(location)
location(self.location)
location(self.location.href)
location(self.location)
location(self)
location(self.href)
location(self)
location(window.location)
location(window.location.href)
location(window.location)
location(window)
location(window.href)
location(window)
self.location = location
self.location = location.href
self.location = location
self.location = self.location
self.location = self.location.href
self.location = self.location
self.location = self
self.location = self.href
self.location = self
self.location = window.location
self.location = window.location.href
self.location = window.location
self.location = window
self.location = window.href
self.location = window
self.location.assign(location)
self.location.assign(location.href)
self.location.assign(location)
self.location.assign(self.location)
self.location.assign(self.location.href)
self.location.assign(self.location)
self.location.assign(self)
self.location.assign(self.href)
self.location.assign(self)
self.location.assign(window.location)
self.location.assign(window.location.href)
```

```
self.location.assign(window.location)
self.location.assign(window)
self.location.assign(window.href)
self.location.assign(window)
self.location.href = location
self.location.href = location.href
self.location.href = location
self.location.href = self.location
self.location.href = self.location.href
self.location.href = self.location
self.location.href = self
self.location.href = self.href
self.location.href = self
self.location.href = window.location
self.location.href = window.location.href
self.location.href = window.location
self.location.href = window
self.location.href = window.href
self.location.href = window
self.location.reload()
self.location.replace(location)
self.location.replace(location.href)
self.location.replace(location)
self.location.replace(self.location)
self.location.replace(self.location.href)
self.location.replace(self.location)
self.location.replace(self)
self.location.replace(self.href)
self.location.replace(self)
self.location.replace(window.location)
self.location.replace(window.location.href)
self.location.replace(window.location)
self.location.replace(window)
self.location.replace(window.href)
self.location.replace(window)
self.location(location)
self.location(location.href)
self.location(location)
self.location(self.location)
self.location(self.location.href)
self.location(self.location)
```

```
self.location(self)
self.location(self.href)
self.location(self)
self.location(window.location)
self.location(window.location.href)
self.location(window.location)
self.location(window)
self.location(window.href)
self.location(window)
self.location = location
self.location = location.href
self.location = location
self.location = self.location
self.location = self.location.href
self.location = self.location
self.location = self
self.location = self.href
self.location = self
self.location = window.location
self.location = window.location.href
self.location = window.location
self.location = window
self.location = window.href
self.location = window
self.location()
self.location(location)
self.location(location.href)
self.location(location)
self.location(self.location)
self.location(self.location.href)
self.location(self.location)
self.location(self)
self.location(self.href)
self.location(self)
self.location(window.location)
self.location(window.location.href)
self.location(window.location)
self.location(window)
self.location(window.href)
self.location(window)
self = location
```

```
self = location.href
self = location
self = self.location
self = self.location.href
self = self.location
self = self
self = self.href
self = self
self = window.location
self = window.location.href
self = window.location
self = window
self = window.href
self = window
self.assign(location)
self.assign(location.href)
self.assign(location)
self.assign(self.location)
self.assign(self.location.href)
self.assign(self.location)
self.assign(self)
self.assign(self.href)
self.assign(self)
self.assign(window.location)
self.assign(window.location.href)
self.assign(window.location)
self.assign(window)
self.assign(window.href)
self.assign(window)
self.href = location
self.href = location.href
self.href = location
self.href = self.location
self.href = self.location.href
self.href = self.location
self.href = self
self.href = self.href
self.href = self
self.href = window.location
self.href = window.location.href
self.href = window.location
```

```
self.href = window
self.href = window.href
self.href = window
self.reload()
self.replace(location)
self.replace(location.href)
self.replace(location)
self.replace(self.location)
self.replace(self.location.href)
self.replace(self.location)
self.replace(self)
self.replace(self.href)
self.replace(self)
self.replace(window.location)
self.replace(window.location.href)
self.replace(window.location)
self.replace(window)
self.replace(window.href)
self.replace(window)
self(location)
self(location.href)
self(location)
self(self.location)
self(self.location.href)
self(self.location)
self(self)
self(self.href)
self(self)
self(window.location)
self(window.location.href)
self(window.location)
self(window)
self(window.href)
self(window)
self = location
self = location.href
self = location
self = self.location
self = self.location.href
self = self.location
self = self
```

```
self = self.href
self = self
self = window.location
self = window.location.href
self = window.location
self = window
self = window.href
self = window
self()
self(location)
self(location.href)
self(location)
self(self.location)
self(self.location.href)
self(self.location)
self(self)
self(self.href)
self(self)
self(window.location)
self(window.location.href)
self(window.location)
self(window)
self(window.href)
self(window)
window.location = location
window.location = location.href
window.location = location
window.location = self.location
window.location = self.location.href
window.location = self.location
window.location = self
window.location = self.href
window.location = self
window.location = window.location
window.location = window.location.href
window.location = window.location
window.location = window
window.location = window.href
window.location = window
window.location.assign(location)
window.location.assign(location.href)
```



```
window.location.assign(location)
window.location.assign(self.location)
window.location.assign(self.location.href)
window.location.assign(self.location)
window.location.assign(self)
window.location.assign(self.href)
window.location.assign(self)
window.location.assign(window.location)
window.location.assign(window.location.href)
window.location.assign(window.location)
window.location.assign(window)
window.location.assign(window.href)
window.location.assign(window)
window.location.href = location
window.location.href = location.href
window.location.href = location
window.location.href = self.location
window.location.href = self.location.href
window.location.href = self.location
window.location.href = self
window.location.href = self.href
window.location.href = self
window.location.href = window.location
window.location.href = window.location.href
window.location.href = window.location
window.location.href = window
window.location.href = window.href
window.location.href = window
window.location.reload()
window.location.replace(location)
window.location.replace(location.href)
window.location.replace(location)
window.location.replace(self.location)
window.location.replace(self.location.href)
window.location.replace(self.location)
window.location.replace(self)
window.location.replace(self.href)
window.location.replace(self)
window.location.replace(window.location)
window.location.replace(window.location.href)
window.location.replace(window.location)
```

```
window.location.replace(window)
window.location.replace(window.href)
window.location.replace(window)
window.location(location)
window.location(location.href)
window.location(location)
window.location(self.location)
window.location(self.location.href)
window.location(self.location)
window.location(self)
window.location(self.href)
window.location(self)
window.location(window.location)
window.location(window.location.href)
window.location(window.location)
window.location(window)
window.location(window.href)
window.location(window)
window.location = location
window.location = location.href
window.location = location
window.location = self.location
window.location = self.location.href
window.location = self.location
window.location = self
window.location = self.href
window.location = self
window.location = window.location
window.location = window.location.href
window.location = window.location
window.location = window
window.location = window.href
window.location = window
window.location()
window.location(location)
window.location(location.href)
window.location(location)
window.location(self.location)
window.location(self.location.href)
window.location(self.location)
window.location(self)
```

```
window.location(self.href)
window.location(self)
window.location(window.location)
window.location(window.location.href)
window.location(window.location)
window.location(window)
window.location(window.href)
window.location(window)
window = location
window = location.href
window = location
window = self.location
window = self.location.href
window = self.location
window = self
window = self.href
window = self
window = window.location
window = window.location.href
window = window.location
window = window
window = window.href
window = window
window.assign(location)
window.assign(location.href)
window.assign(location)
window.assign(self.location)
window.assign(self.location.href)
window.assign(self.location)
window.assign(self)
window.assign(self.href)
window.assign(self)
window.assign(window.location)
window.assign(window.location.href)
window.assign(window.location)
window.assign(window)
window.assign(window.href)
window.assign(window)
window.href = location
window.href = location.href
window.href = location
```

```
window.href = self.location
window.href = self.location.href
window.href = self.location
window.href = self
window.href = self.href
window.href = self
window.href = window.location
window.href = window.location.href
window.href = window.location
window.href = window
window.href = window.href
window.href = window
window.reload()
window.replace(location)
window.replace(location.href)
window.replace(location)
window.replace(self.location)
window.replace(self.location.href)
window.replace(self.location)
window.replace(self)
window.replace(self.href)
window.replace(self)
window.replace(window.location)
window.replace(window.location.href)
window.replace(window.location)
window.replace(window)
window.replace(window.href)
window.replace(window)
window(location)
window(location.href)
window(location)
window(self.location)
window(self.location.href)
window(self.location)
window(self)
window(self.href)
window(self)
window(window.location)
window(window.location.href)
window(window.location)
window(window)
```

```
window(window.href)
window(window)
window = location
window = location.href
window = location
window = self.location
window = self.location.href
window = self.location
window = self
window = self.href
window = self
window = window.location
window = window.location.href
window = window.location
window = window
window = window.href
window = window
window()
window(location)
window(location.href)
window(location)
window(self.location)
window(self.location.href)
window(self.location)
window(self)
window(self.href)
window(self)
window(window.location)
window(window.location.href)
window(window.location)
window(window)
window(window.href)
window(window)
```

Quelle:

www.phpied.com/files/location-location/location-location.html

Leuchtdioden (LEDs) mit HTML / CSS anzeigen

geschrieben von Andreas Potthoff | 30. Juli 2017

Hier eine Möglichkeit LEDs mit HTML und CSS zu realisieren:



HTML

Die HTML Container für die Anzeige der jeweiligen LED-Farbe:

```
<div class="led-red"></div>
<div class="led-yellow"></div>
<div class="led-green"></div>
<div class="led-blue"></div>
<div class="led-grey"></div>
```

CSS

Die entsprechende CSS-Datei für die LEDs:

```
.led-red {
  margin: 5px auto;
  width: 12px;
  height: 12px;
  background-color: #940;
  border-radius: 50%;
  box-shadow: #000 0 -1px 7px 1px, inset #600 0 -1px 9px,
  #F00 0 2px 12px;
}
```

```
.led-yellow {  
    margin: 5px auto;  
    width: 12px;  
    height: 12px;  
    background-color: #A90;  
    border-radius: 50%;  
    box-shadow: #000 0 -1px 7px 1px, inset #660 0 -1px 9px,  
#DD0 0 2px 12px;  
}
```

```
.led-green {  
    margin: 5px auto;  
    width: 12px;  
    height: 12px;  
    background-color: #690;  
    border-radius: 50%;  
    box-shadow: #000 0 -1px 7px 1px, inset #460 0 -1px 9px,  
#7D0 0 2px 12px;  
}
```

```
.led-blue {  
    margin: 5px auto;  
    width: 12px;  
    height: 12px;  
    background-color: #4AB;  
    border-radius: 50%;  
    box-shadow: #000 0 -1px 7px 1px, inset #006 0 -1px 9px,  
#06F 0 2px 14px;  
}
```

```
.led-grey {  
    margin: 5px auto;  
    width: 12px;  
    height: 12px;  
    background-color: #C0C0C0;  
    border-radius: 50%;  
    box-shadow: #000 0 -1px 7px 1px, inset #9A9 0 -1px 9px,  
#969 0 2px 14px;  
}
```

Die Farbe grau habe ich selbst erstellt.

Die CSS-Datei im HTML Header laden:

```
<head>
....
<link rel="stylesheet" href="path_to_your_css/led.css">
</head>
```

- Autor: devilsbackyard
 - License: Open Source
 - Quelle: cssdeck.com/labs/css-leds
-

Cryptobörse: Wie kann ich in Python den Ticker der Push API (WAMP) von Poloniex abonnieren?

geschrieben von Andreas Potthoff | 30. Juli 2017

Da ich gerade eine Trader-Applikation für Poloniex in Python schreibe, wollte ich dazu als erstes den Ticker der Push API abonnieren. Ich habe dazu das Modul Autobahn in Python installiert und den folgenden Code verwendet. Hier wird der gesamte Ticker-Feed über das WAMP-Protokoll des Websockets abonniert und die empfangenen Daten (Events) werden ausgegeben. Das klappt prima!

Getestet mit: Python 3.5.2 @ Ubuntu 16.04 LTS

Code:

```
from autobahn.asyncio.wamp import ApplicationSession
```



```

from autobahn.asyncio.wamp import ApplicationRunner
from asyncio import coroutine

class PoloniexComponent(ApplicationSession):
    def onConnect(self):
        self.join(self.config.realm)

    @coroutine
    def onJoin(self, details):
        def onTicker(*args):
            print("Ticker event received:", args)

        try:
            yield from self.subscribe(onTicker, 'ticker')
        except Exception as e:
            print("Could not subscribe to topic:", e)

def main():
    runner = ApplicationRunner("wss://api.poloniex.com:443",
                               "realm1")
    runner.run(PoloniexComponent)

if __name__ == "__main__":
    main()

```

Ausgabe:

```

Ticker event received: ('ETH_ETC', '0.14829502', '0.14992249',
'0.14829510', '-0.04851949', '12292.09070919',
'80914.07777792', 0, '0.15614805', '0.14798000')
Ticker event received: ('USDT_ETC', '1.69796894',
'1.69796456', '1.68289334', '-0.00495273', '50045.81062071',
'29268.11748844', 0, '1.74392238', '1.68000000')
Ticker event received: ('BTC_CLAM', '0.00103996',
'0.00103996', '0.00102894', '0.02329059', '6.67321322',
'6532.51214055', 0, '0.00106603', '0.00099600')
Ticker event received: ('BTC_XCN', '0.00000257', '0.00000257',
'0.00000251', '-0.04104477', '8.59036050', '3316114.77166474',
0, '0.00000280', '0.00000250')

```

Ticker event received: ('USDT_LTC', '3.64300000', '3.66651211', '3.64300000', '0.00192519', '2495.39248941', '688.99598487', 0, '3.67999050', '3.59000000')

Ticker event received: ('USDT_NXT', '0.02430000', '0.02629471', '0.02400000', '-0.08963942', '3555.89759389', '136298.49703595', 0, '0.02774462', '0.02430000')

Ticker event received: ('XMR_HYP', '0.00009300', '0.00010314', '0.00009000', '-0.15462230', '23.89571026', '241095.35982673', 0, '0.00011499', '0.00009300')

Ticker event received: ('XMR_NXT', '0.00732377', '0.00779997', '0.00719002', '-0.35367360', '143.71435777', '15777.31974579', 0, '0.01133138', '0.00732377')

Ticker event received: ('BTC_ETH', '0.01923009', '0.01923250', '0.01923009', '0.00972015', '3762.76051615', '195213.35234572', 0, '0.01950000', '0.01902000')

Ticker event received: ('BTC_DAO', '0.00019103', '0.00019306', '0.00019107', '-0.00639758', '37.37047925', '194598.39669886', 0, '0.00019492', '0.00018900')

Ticker event received: ('ETH_STEEM', '0.12000000', '0.12771693', '0.11820535', '-0.05064845', '1131.24971868', '9017.54958625', 0, '0.13468721', '0.12000000')

Ticker event received: ('BTC_ETC', '0.00286215', '0.00286206', '0.00286203', '-0.02824461', '2723.26141525', '927741.49933883', 0, '0.00300899', '0.00285000')

Ticker event received: ('ETH_ETC', '0.14829502', '0.14992247', '0.14829511', '-0.04851949', '12292.09070919', '80914.07777792', 0, '0.15614805', '0.14798000')

Ticker event received: ('USDT_ETC', '1.69796894', '1.69796453', '1.68289334', '-0.00495273', '50045.81062071', '29268.11748844', 0, '1.74392238', '1.68000000')

Ticker event received: ('BTC_XMR', '0.00554600', '0.00556799', '0.00555000', '0.24788213', '6990.80381868', '1412686.14372726', 0, '0.00557071', '0.00427288')

Ticker event received: ('USDT_LTC', '3.64300000', '3.66651208', '3.64300000', '0.00192519', '2495.39248941', '688.99598487', 0, '3.67999050', '3.59000000')

Ticker event received: ('BTC_ETH', '0.01923009', '0.01923226', '0.01923009', '0.00972015', '3762.76051615', '195213.35234572', 0, '0.01950000', '0.01902000')

Ticker event received: ('BTC_FCT', '0.00429347', '0.00431739', '0.00429346', '0.03462842', '1408.22052281',

```
'340507.08137862', 0, '0.00440000', '0.00381938')
Ticker event received: ('BTC_VOX', '0.00007348', '0.00007419',
'0.00007341', '-0.00284977', '171.75237449',
'2366663.48717554', 0, '0.00007749', '0.00006606')
Ticker event received: ('BTC_DAO', '0.00019103', '0.00019306',
'0.00019108', '-0.00639758', '37.37047925', '194598.39669886',
0, '0.00019492', '0.00018900')
Ticker event received: ('ETH_LSK', '0.02715264', '0.02715264',
'0.02609935', '0.00565333', '1535.89611701', '56262.64389120',
0, '0.02800000', '0.02587547')
Ticker event received: ('ETH_ETC', '0.14829502', '0.14992245',
'0.14829511', '-0.04851949', '12292.09070919',
'80914.07777792', 0, '0.15614805', '0.14798000')
Ticker event received: ('USDT_ETC', '1.69796894',
'1.69796451', '1.68289334', '-0.00495273', '50045.81062071',
'29268.11748844', 0, '1.74392238', '1.68000000')
```

Die entsprechenden Labels für diese Daten sind in der Reihenfolge:

currencyPair, last, lowestAsk, highestBid, percentChange, baseVolume, quoteVolume, isFrozen, 24hrHigh, 24hrLow

Jetzt dürfte es kein Problem mehr sein diese Daten mit Python weiterzuverarbeiten.

Commodore Geschichte: Doku: 8-Bit Generation – The Commodore Wars (2016, engl.)

geschrieben von Andreas Potthoff | 30. Juli 2017