VS2015 scheint nicht zuverlässig erkennen zu können, welche Dateien sich im Projekt geändert haben und welche nicht. Bei (De-)/Installation eines npm-Pakets z.B. kann es sein, das VS eine neues Deployment Layout erstellt und somit alle Dateien (inkl. node\_modules) erneut kopiert. Dies dauert oftmals mehrere Minuten!

Eine Dokumentation oder ähnliches, unter welchen Bedingungen VS ein neues Layout erstellt (also ALLE Dateien löscht und neu kopiert), bzw. wie sich dieses Verhalten beeinflussen lässt, wäre sehr hilfreich.

Um ein komfortables Entwickeln und Debuggen zu ermöglichen sind folgende Schritte notwendig:

1. Installation von Chakra Node (Node mit ChakraCore) auf dem Device
   1. <https://github.com/nodejs/node-chakracore/releases>
   2. Downloads -> ARM-Packages
   3. Entpacken z.B.: nach ’C:\Node (Chakra)’
2. Hinzufügen des Node-Installationsverzeichnisses zu PATH (**permanent**)
   1. Prüfen der aktuellen PATH-Variable mit:
      1. $ENV:PATH
   2. Auslesen der aktuellen PATH-Variable aus Registry:
      1. (Get-ItemProperty -Path ‘Registry::HKEY\_LOCAL\_MACHINE\System\CurrentControlSet\Control\Session Manager\Environment’ -Name PATH).path
   3. Wert in Variable merken für später:
      1. $oldpath = (Get-ItemProperty -Path ‘Registry::HKEY\_LOCAL\_MACHINE\System\CurrentControlSet\Control\Session Manager\Environment’ -Name PATH).path
   4. Variable um neuen Pfad (Node Chakra Verezeichnis) ergänzen:
      1. $newpath = “$oldpath;c:\path\to\folder”
   5. Neue Pfad-Variable in Registry persistieren:
      1. Set-ItemProperty -Path ‘Registry::HKEY\_LOCAL\_MACHINE\System\CurrentControlSet\Control\Session Manager\Environment’ -Name PATH –Value $newPath
   6. Überprüfen der neuen Pfad-Variable:
      1. (Get-ItemProperty -Path ‘Registry::HKEY\_LOCAL\_MACHINE\System\CurrentControlSet\Control\Session Manager\Environment’ -Name PATH).path
   7. PS-Session neu starten und prüfen ob Node von beliebigem Ordner ausgeführt warden kann:
      1. node –v
3. Einmaliges Deployment der App mit Visual Studio und Deployment-Verzeichnis merken
   1. z.B.: c:\data\users\DefaultAccount\AppData\Local\Packages\NavigatorWeb\_8f0qdkcs9hxtj\LocalState
4. Deployment-Verzeichnis auf dem Device freigeben:
   1. net share DeploymentFolder=c:\data\users\DefaultAccount\AppData\Local\Packages\NavigatorWeb\_8f0qdkcs9hxtj\LocalState /GRANT:Everyone`,FULL /REMARKS:”Navigator Web Deployment Folder”
5. Deployment-Verzeichnis auf Entwicklungsrechner als Netzlaufwerk einrichten (optional): z.B.: W:\
6. Command Task Runner Extension in Visual Studio installieren und TaskRunnerExplorer öffnen
7. Die Datei commands.json im Solution-Verzeichnis ‘scripts’ öffnen und editieren:
   1. Der hier definierte ‘Deploy’-Task ruft ein PowerShell Skript auf, welches mittels robocopy alle Projektdateien in das Zielverzeichnis kopiert
   2. Dazu muss das Zielverzeichnis entsprechend angepasst warden:
      1. -Dest \"W:/\" 🡪 **W:/** ersetzen durch den Pfad des gemappten Netzlaufwerks oder dem Netzwerkpfad zum Deploymentverzeichnis
8. Das Projekt wird von nun an automatisch deployed wen nein Build erzeugt wird
9. Zum Debuggen der Applikation
   1. Device 🡪 Enter-PSSession
   2. Navigieren zum Deployment-Verzeichnis, e.g.: c:\data\users\DefaultAccount\AppData\Local\Packages\NavigatorWeb\_8f0qdkcs9hxtj\LocalState
   3. Ausführen von: node ./bin/www