



Inhaltsverzeichnis

1. Einführung.....	2
1.1 Was ist ioBroker.....	2
1.2 Wofür ist ioBroker gedacht.....	2
1.3 Wie ist ioBroker aufgebaut.....	2
1.4 Was ist nodeJS und npm.....	2
1.5 Struktur nodeJS und ioBroker.....	3
2. Vorbereitungen.....	4
2.1 git installieren.....	4
2.2 Template Adapter herunterladen.....	5
2.3 Erklärung des Templates.....	5
3. IDE - Webstorm.....	7
3.1 Webstorm einrichten.....	7
3.2 Projekt auf Github Veröffentlichen.....	7
4. Adapter auf npm Veröffentlichen oder updaten.....	7
5. Links.....	7



1. Einführung

1.1 Was ist ioBroker

ioBroker ist eine sogenannte Middleware (Integrationsplattform) die es ermöglichen soll jede Hardware und Software Lösung miteinander zu verbinden. Sie ist Vollständig in JavaScript (**nicht** JAVA) geschrieben.

1.2 Wofür ist ioBroker gedacht

Ursprünglich war ioBroker dafür gedacht Smart Home Geräte mit einander zu verbinden, Steuern und Visualisieren. Mittlerweile beschränkt sich der Funktionsumfang, dank der Adapter, nicht mehr nur auf Smart Home Geräte. Es ist möglich alles denkbare mit ioBroker zu verbinden und zu Steuern, wenn es irgendeine Art von Schnittstelle hat.

1.3 Wie ist ioBroker aufgebaut

Es handelt sich um ein Modulares System in dem viele Komponenten unabhängig von einander arbeiten. Die meisten Adapter sind Optional und für den Betrieb von ioBroker nicht notwendig.

Zwingend nötig für den Betrieb von ioBroker sind der js-controller und der Admin Adapter.

Es gibt weitere Adapter wie VIS die Abhängigkeiten haben, diese werden dann automatisch mit installiert.

ioBroker wird von dem Programm nodeJS ausgeführt. Jeder Adapter wird als eigenständiger Prozess gestartet, das erhöht die Stabilität des Systems, da ein Fehler in einem Adapter nur diesen zum Absturz bringt.

1.4 Was ist nodeJS und npm

nodeJS ist ein Interpreter der JavaScript in Echtzeit interpretiert. Er basiert auf „V8“ einer Laufzeitumgebung die für den Chrome Browser entwickelt wurde.

Ursprünglich kam JavaScript fast ausschließlich in Browsern zum Einsatz, weshalb bei der Entwicklung auf sehr viel Hilfe aus dem Bereich Webentwicklung zurückgegriffen werden kann.

Erst mit der Entwicklung von nodeJS wurde JavaScript auch in anderen Anwendungen großflächig eingesetzt. Im Gegensatz zu der Verwendung im Browser ermöglicht nodeJS auch die direkte Interaktion mit der Hardware.

npm ist eine Paketverwaltung die es ermöglicht Module und all ihre Abhängigkeiten mit nur einem Befehl zu installieren. Dazu muss man keine URL kennen oder irgendwelche Zip Dateien herunterladen, es reicht den Namen des Moduls zu kennen. Es ist auch möglich gezielt eine bestimmte Version eines Moduls zu installieren.

ioBroker verwendet ebenfalls npm zur Installation und Verwaltung der Adapter, davon bekommt man als Benutzer jedoch wenig mit und auch als Entwickler muss man sich relativ wenig damit beschäftigen. Am meisten hat man damit zu tun wenn man seinen Adapter veröffentlicht, dazu später mehr.



1.5 Struktur nodeJS und ioBroker

nodeJS ist die Basis für ioBroker und führt ioBroker aus. Jeder Adapter von ioBroker wird als Eigenständiger Prozess gestartet.

nodeJS stellt alle Notwendigen Funktionen bereit um mit der Hardware zu Kommunizieren.

Der js-controller ist das Herz von ioBroker, er verwaltet die Adapter, Objekte und regelt die Kommunikation zwischen den Adaptern. Außerdem stellt er Funktionen bereit die von den meisten Adaptern benötigt werden, das vereinfacht die Adapter Entwicklung.

Der Admin Adapter stellt die Grafische Oberfläche zur Verfügung die es dem Benutzer ermöglicht ioBroker zu Verwalten.



2. Vorbereitungen

- ✓ [Github.com](https://github.com) Account anlegen
- ✓ npmjs.com Account anlegen
- ✓ Entwicklungsumgebung: [Webstorm](https://www.jetbrains.com/webstorm/) von Jet Brains installieren. Es kann auch jede andere Entwicklungsumgebung verwendet werden, aber in diesem HowTo wird Webstorm verwendet.
- ✓ IoBroker auf dem selben PC installieren wie Webstorm

2.1 git installieren

Debian/Ubuntu:

In der Konsole „sudo apt-get install git“ ausführen

Windows:

Auf die Webseite <http://msysgit.github.com/> gehen und die zum

Download angebotene Version herunterladen und installieren.

Mac OS X:

Von der Seite <http://sourceforge.net/projects/git-osx-installer/>

den Installer herunterladen und ausführen.



2.2 Template Adapter herunterladen

1. In der Konsole/Eingabeaufforderung in das Verzeichniss „iobroker/node_modules“ wechseln und „git clone --depth=1 https://github.com/ioBroker/ioBroker.template.git“ ausführen.
Jetzt wird der gesamte Template Adapter von Github heruntergeladen und im Ordner „iobroker.template“ gespeichert.
2. Abhängigkeiten installieren die zur Entwicklung benötigt werden:
„npm install“
im Ordner „iobroker.template“ ausführen
3. Adapter Name und Entwickler infomationen für den Adapter festlegen. Dazu folgenden Befehl (name, email und author anpassen) ausführen:
„gulp rename --name mynewname --email email@mail.com --author "Author Name" „
Template Ordner, „iobroker.template“, Umbennen in „iobroker.mynewname“.
4. „git“ Ordner entfernen:
Windows Konsole → „del .git“
Linux Konsole → „rm -r .git“

2.3 Erklärung des Templates

- admin – i18n: Hier liegen die Texte mit Übersetzungen die in der Adapter Konfiguration angezeigt werden. Daraus wird die Datei words.js erzeugt.
- admin – index.html: Diese Datei wurde für die Anzeige der Adapterkonfiguration verwendet, sie wird seit Einführung von Admin in der Version 3 nicht mehr verwendet.
- admin – index_m.html: Nachfolger der index.html. Hier erstellt der Adapter Entwickler die Grafische Oberfläche zur Adapterkonfiguration. Dafür wird HTML, JS und das CSS Framework MaterializeCSS verwendet. Um einfach neue Tabs innerhalb der Adapterkonfiguration zu erzeugen findet man auf <http://iobroker.net:3000/> ein Tool.
- admin – template.png: Das ist das Logo das in der Adapterkonfiguration, dem Reiter Adapter und Instanzen angezeigt wird.
- admin – words.js: Darin befinden sich alle Übersetzungen zu den angezeigten Texten in der Adapterkonfiguration. Zur vereinfachten Übersetzung findet man auf <http://iobroker.net:3000/> ein Tool.
- docs - *: In dem Ordner docs und seinen Unterordnern erstellt man die Dokumentation/das Handbuch und übersetzt es in mehrere Sprachen. Dieses System ersetzt die Dokumentation in der Datei readme.MD, damit in der Zukunft Automatisiert ein Großes Benutzerhandbuch an Zentraler stelle erzeugt werden kann.
- lib – utils.js: Die utils.js stellt grundlegende Funktionen von ioBroker für den Adapter bereit.
- lib: Im Ordner lib kann man eigene Funktionsbibliotheken ab legen die im Adapter verwendet werden sollen.
- test: Im Ordner Test liegen Skripte die bestimmte Funktionstest durchführen. Für diese Tests wird die Plattform Travis-CI verwendet. Dank Appollon77 muss man hier als Entwickler sehr wenig bis gar nichts machen. Wenn dazu fragen bestehen am besten direkt Appollon77 Kontaktieren, der hat die Tests erstellt und kennt sich gut aus.
- widgets: In diesem Ordner kann man Widgets für VIS ablegen die mit dem Adapter zusammen installiert werden. Wenn man keine Widgets bereit stellt kann der Ordner gelöscht werden.



- `www`: In diesem Ordner kann man eine Webseite ablegen, die dann vom WEB Adapter bereit gestellt wird. Wenn keine Webseite mit dem Adapter geliefert werden soll, kann man den Ordner einfach löschen.
- `.gitignore`: Diese Datei enthält eine Liste mit Dateien die von ‚git‘ nicht verwaltet werden sollen und somit von der History ausgeschlossen sind.
- `.npmignore`: Diese Datei enthält eine Liste mit Dateien die von ‚npm‘ nicht installiert werden sollen.
- `.travis.yml`: Diese Datei gehört zu den Travis-CI Tests. In ihr sind grundlegende Informationen für die Tests definiert.
- `LICENSE`: Die Lizenz des Adapters legt fest wie der Adapter verwendet werden darf und damit auch der Code.
- `README.md`: Hier kommt eine Kurze Beschreibung des Adapters hinein, bevorzugt in Englisch. Außerdem wird auch das Changelog geführt.
- `appveyor.yml`: Diese Datei gehört zu den Appveyor Tests. In ihr sind grundlegende Informationen für die Tests definiert.
- `gulpfile.js`: Gulp ist ein Konsolen Tool mit dessen Hilfe bestimmte Funktionen ausgeführt werden können. Diese Funktionen sind in der Datei `gulpfile.js` definiert. Die Funktionen sind speziell für die Entwicklung von ioBroker Adapter geschrieben. Zurzeit nur zur Erzeugung der Datei `words.js`.
- `io-package.json`: Die `io-package.json` ist eine an die `package.json` angelehnte Version die für die Adapterverwaltung in ioBroker verwendet wird. Hier wird auch ein Changlog geführt das im Admin als Tooltip angezeigt wird, es ermöglicht auch die Installation früherer Adapter versionen.
- `main.js`: Hier kommt der eigentliche Code des Adapters hinein.
- `package.json`: In der `package.json` werden grundlegende Informationen für npm definiert.



3. IDE - Webstorm

3.1 Webstorm einrichten

1. Webstorm starten
2. „Open“ → den Adapter Ordner suchen der unter 2.2 Umbenannt wurde (iobroker/node_modules/)
3. Run → Edit Configurations...
4.  → „Node.js“
5. Name: iobroker.meinAdapter
JavaScript file: main.js
Application parameters: --force

3.2 Projekt auf Github Veröffentlichen

1. GitHub Account hinterlegen:
File → Settings → Version Control → GitHub
2. GitHub repository anlegen:
VCS → Import into Version Control → Share Project on GitHub
3. New repository name: muss mit „ioBroker.“ anfangen
Remote name: origin

4. Adapter auf npm Veröffentlichen oder updaten

Als erstes in der Konsole/Eingabeaufforderung den folgenden Befehl ausführen
„*npm login*“

a) aus den Lokalen sourcen:

In der Konsole/Eingabeaufforderung den folgenden Befehl im Adapter Ordner ausführen
„*npm publish*“

b) aus den GitHub sourcen:

In der Konsole/Eingabeaufforderung den folgenden Befehl ausführen
„*npm publish https://github.com/deinBenutzername/ioBroker.deinAdapter*“

5. Links

1. JavaScript/Webtechnik Nachschlagewerk → <https://www.w3schools.com>
2. Adapter Dev Doku → <https://github.com/ioBroker/ioBroker/wiki/Adapter-Development-Documentation>
3. npm Publish oder update → <https://docs.npmjs.com/getting-started/publishing-npm-packages>
4. Developer Tools für ioBroker → <http://iobroker.net:3000/>
5. Konsolen Befehle ioBroker → <https://github.com/ioBroker/ioBroker/wiki/Console-commands>
6. Regular Expression online Tool → <https://regex101.com/>
7. nodeJS → <https://de.wikipedia.org/wiki/Node.js>
8. npm → [https://de.wikipedia.org/wiki/NPM_\(Software\)](https://de.wikipedia.org/wiki/NPM_(Software))

