

# ● Domotica: het Internet der Dingen ●

Gjalt Zwaagstra

CompUsers is bezig met een start en het is de bedoeling dat alle kennis over Domotica én IOT (Internet of Things) in de Domotica-IG zijn terug te vinden. Als kenniscentrum kunnen wij als HCC!domotica/IoT-IG ook weer interessant worden voor niet-HCC-ers. Maar ... dat is niet eenvoudig.

We moeten daarvoor eerst de kennis en vaardigheden bijeenbrengen aan onze kernleden, om deze vervolgens te kunnen aanbieden aan onze leden.

## Kennis en kosten

Met kennis van domotica kun je allerlei problemen oplossen, maar tegen welke kosten?

### Hoe kun je domotica zoal voor je huis inzetten?

- Automatiseren van licht en kachel aan als je thuis bent.
- Alarm aan als er niemand thuis is.
- Elektrische deken inschakelen voor het slapen.
- Buitenverlichting op zonnetijden schakelen.
- Film/tv-series streamen zodra er een nieuwe aflevering is.
- Wekkerradio integreren in het systeem.
- Zonwering geheel automatisch laten werken.
- Gordijnen automatisch sluiten bij schemer en openen in de ochtend.
- En uiteraard voor senioren: 'slimme zorg'.

Auteur: Tom de Hoog

## Slimme domotica maakt het verschil in de zorg

**H**et concept voor de zorgdomotica in De Lindewoord is afkomstig van technisch dienstverlener en systeemintegrator Leertouwer. Dit bedrijf levert onder de naam Conviv Care geïntegreerde en merkonafhankelijke domotica-oplossingen voor de zorgsector. Jasper Coppes is als projectmanager bij Leertouwer nauw betrokken bij de ontwikkeling en implementatie van Conviv Care. "We onderzochten zo'n vier jaar terug hoe we met Conviv Care de zorgmarkt beter konden bedienen. Bij Leertouwer hadden we al veel techniek en kennis in huis en we wilden graag verbinding maken tussen de technologie die wij aanbieden en de vraag in de zorg. We merkten door ons onderzoek dat de technologie die men tot dan toe paste beperkt was tot alarmering en oproepsystemen. Zorgorganisaties willen echter graag innoveren en daar springen veel fabrikanten op in door continue nieuwe producten te ontwikkelen. Daardoor zie je echter systemen met steeds meer en complexere functionaliteit. Dat sluit naar onze mening niet goed aan bij de wensen uit de zorgsector. Met ons doorontwikkelde Conviv Care met dwaaldetectie, verpleegproepsystemen en camera-systemen leveren we een merkonafhankelijke oplossing met een gebruiksvriendelijke interface. Bedienen doe je als gebruiker via diverse schermen: desktop, tablet of smartphone."

**Eerst luisteren dan oplossingen bieden**  
Coppes: "Wij doen in eerste instantie consultancy om de verbinding te leggen met de wensen van de zorgverleners, de wensen van cliënten en de vastgoed-



## Welke systemen zijn er?

- Klik-aan-Klik-uit
- Zigbee, Z-wave
- KNX
- App op je smartphone met de aanschaf van slimme componenten.
- Wifi (iedereen heeft dit in huis, wie niet...?)

## Eigen systeem of lekker makkelijk in de Cloud.

Een Cloudsysteem werkt snel, is eenvoudig te configureren via een app op je telefoon en werkt spraakgestuurd met een slimme speaker. Componenten zijn meestal via wifi of een bridge aan te sturen.

Voordeel: goedkoop in aanschaf. Nadeel: je betaalt met je privacy. Jouw gegevens gaan naar de provider in de cloud (o.m. China), Google en Amazon. Hierdoor krijgen ze jouw

slimme woning met je (adres)gegevens perfect in kaart. En ze verzamelen alle logdata. Daardoor kunnen ze voor jouw situatie prima aanbiedingen maken.

## Eigen systeem programmeren

Je kunt zelf een systeem inrichten en goedkope schakelaars en lampen zelf flashen. Het alternatief is: je koopt kant-en-klare apparaten (veel duurder, vaak de dubbele prijs!). **Voordeel** van het zelf opbouwen: jouw gegevens gaan niet naar de cloud en blijven binnenshuis. **Nadeel**: je hebt kennis van zaken nodig. Met HCC haal je die kennis nu in huis.

## Is er een standaard Domoticasysteem?

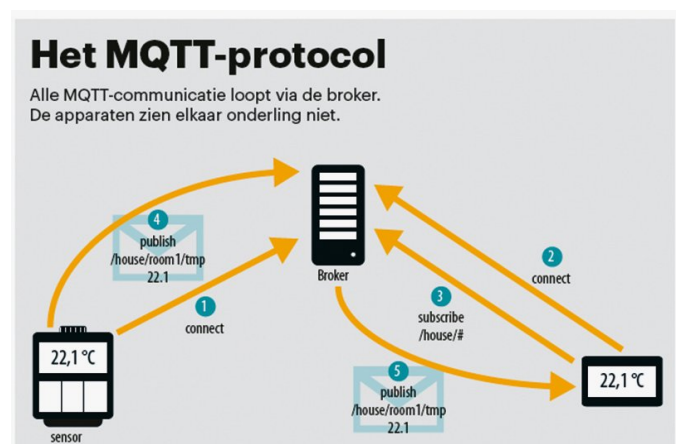
Er zijn systemen zoals Domoticz, Homey, HomeAssistant, Philips Hue, Ikea en Action. Er is heel veel te koop, maar wat is nu *slim* voor je slimme woning? Er is standaardisering in Domoticaland; die heet 'Internet der dingen (IOT).'

Welk protocol is hiervoor ontwikkeld: MQTT, dat staat voor 'Message Queueing Telemetry Transport'. Wat houdt dat in?

Message Queueing Telemetry Transport (MQTT) is een machine-tot-machine (M2M) datatransferprotocol dat in een hoog tempo het leidende messaging-protocol voor het Industriële Internet der Dingen (IIoD) is geworden.

## Wat is MQTT en waarom is het ideaal?

De nieuwste versie, MQTT 3.1.1, is een OASIS-standaard die open en royaltyvrij is. OASIS is de *Organization for the Advancement of Structured Information Standard*, een internationaal consortium dat de goedkeuring voor productonafhankelijke standaarden voor informatie bevordert.



Het is lichtgewicht en dus ideaal voor het monitoren op afstand, vooral bij M2M-verbindingen waarvoor een kleine footprintcode vereist is of waar de netwerkbandbreedte beperkt is.

Message Queueing Telemetry Transport werd in 1999 bedacht door dr. Andy Stanford-Clark en Arlen Nipper. Mede-uitvinder Arlen Nipper is de president van Cirrus Link Solutions, het bedrijf dat de Cirrus Link MQTT-modules voor Ignition heeft ontwikkeld.

## Hoe werkt MQTT?

MQTT is een publish/subscribe-protocol dat edge-of-netwerkapparaten in staat stelt te publiceren naar een broker. Cliënten maken verbinding met deze broker, die vervolgens bemiddelt in de communicatie tussen de twee apparaten.

Elk apparaat kan inschrijven of registreren voor bepaalde onderwerpen. MQTT is bidirectioneel en ondersteunt stateful session-awareness. Als een edge-of-netwerkapparaat de verbinding verliest, worden alle geabonneerde cliënten geïnformeerd door de 'Last Will and Testament'-functie van de MQTT-server. Daardoor kan iedere geautoriseerde cliënt in het systeem een nieuwe waarde terugpubliceren naar het edge-of-netwerkapparaat.

## Protocol met 3 QoS-klassen

MQTT heeft een mechanisme voor Quality of Service (QoS). Daarbij gaat het er niet om dat bepaalde berichten voorrang krijgen, maar om een soort ontvangstbevestiging. Elk bericht dat verstuurd wordt, krijgt een QoS-niveau mee.

**Niveau 0** bepaalt dat een bericht zonder bevestiging eenmalig verstuurd wordt. Dat gaat snel en bespaart op bronnen. Voor eenvoudige sensorwaarden is dat voldoende, maar het is niet aan te bevelen als het functioneren van een fabrieksinstallatie afhankelijk is van die informatie en de netwerkverbinding slecht is.

Een bericht met de **QoS-waarde 1** komt minstens één keer aan bij de ontvanger. Als antwoord op een PUBLISH verstuurt de ontvanger PUBACK. Als deze reactie uitblijft, probeert de zender het opnieuw tot de ontvangst bevestigd wordt. Bij die procedure kan het gebeuren dat een bericht vaker aankomt als de bevestiging verloren gaat. De ontwikkelaar moet ervoor zorgen dat daar geen problemen door kunnen ontstaan. Als een robotarm per ongeluk meerdere keren de opdracht krijgt een bepaalde afstand naar voren te gaan, kan dat vervelende gevolgen hebben.

**QoS-niveau 2** van het MQTT-protocol is ingewikkelder, maar ook betrouwbaarder. Daarbij dragen beide gesprekspartners er zorg voor dat een bericht precies één keer bij de partner aankomt. Zender A verzendt een bericht via PUBLISH met een bericht-ID. Ontvanger B bevestigt dit met PUBREC en slaat het bericht zolang op. Als zender A deze bevestiging ontvangt, retourneert hij PUBREL, waarna hij het bericht gerust kan schrappen. Hij zal het onder geen voorwaarde opnieuw verzenden. Hij weet nu zeker dat B het bericht ontvangen heeft. Ten slotte stuurt ontvanger B een laatste bericht naar A: PUBCOMP. Pas dan begint B het bericht te verwerken.

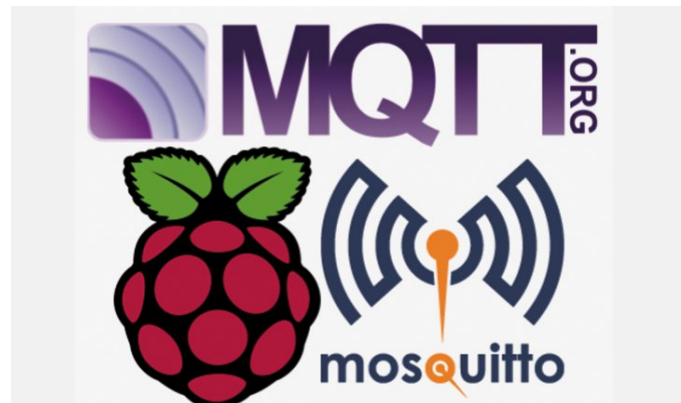
Als B een broker is zal hij pas na het zenden van PUBCOMP het bericht naar de abonnees verzenden. Als een van de twee partners een ontvangstbevestiging ontvangt buiten de gedefinieerde wachttijd, gaat hij een stap terug en verzendt hij zijn laatste bevestiging nog een keer, maar nooit nogmaals het oorspronkelijke bericht.

## Offline bericht

Een verbroken verbinding komt vaker voor dan je denkt. Als een apparaat offline gaat, kan het niet meer alle apparaten waarschuwen die misschien op een bericht wachten. De ontwikkelaars van MQTT hebben in het protocol met die situatie rekening gehouden.

Wanneer een client verbinding maakt met de broker, kan hij een 'laatste wil' deponeren en daarvoor dezelfde eigenschappen instellen die ook voor normale berichten van het protocol gelden: topic, payload, QoS en retain-vlag.

De broker accepteert die laatste wil en geeft hem door aan de abonnees als de verbinding verloren gaat. Installatie van de broker op een Raspberry Pi onder het standaard Raspbian-OS gaat met het volgende commando:



```
sudo apt-get install -y mosquitto mosquitto-clients << er
wordt een complete installatie uitgevoerd.
Het commando: sudo systemctl enable mosquitto.service <<
zal de broker starten bij een boot.
```

## De MQTT berichten, hoe maak je die?

De echte domoticasystemen hebben een MQTT-add-on die je alleen maar aan hoeft te zetten. En dan kunnen ze met de broker praten en luisteren.

Zo stuur je MQTT-berichten de hele wereld over om iets in je slimme woning voor elkaar te krijgen.

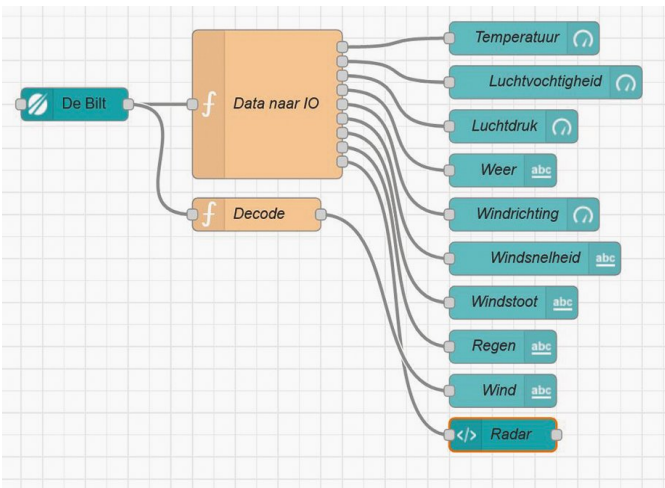
```
25-4-2020 22:26:16node: 20ebff06.489338
msg.payload: Object
object
buienradar: object
sunrise: "2020-04-25T06:20:00"
sunset: "2020-04-25T20:55:00"
radarurl:
"https://api.buienradar.nl/image/1.0/RadarMapNL?w=
500&h=512"
stationcode: 6269
stationnaam: "Meetstation Lelystad"
temperatuurGC: 8
windsnelheidBF: 2.6
luchtvochtigheid: 76
datum: "2020-04-25T22:20:00"
zichtmeters: 42500
icoonactueel:
https://www.buienradar.nl/resources/images/icons/
weather/30x30/aa.png"
icoonzin: "Vrijwel onbewolkt (zonnig/helder)"
regenMMPU: 0
regenMM24U: 0
luchtdruk: 1014
windrichtingGR: 6
zonintensiteitWM2: 0
windsnelheidMS: 2.6
windstotenMS: 4.7
```

## U wilt de hele wereld managen vanuit huis.

Met dit systeem kunt u gegevens binnenhalen. Denk aan weerstations, virusmeldingen, vluchtgegevens, filemeldingen, radio- en tv-streams en alles van IoT. Hiermee bent u klaar voor de toekomst.

### Node Red

Node-RED (NR) is een applicatie waarmee je met weinig programmeerervaring krachtige automatiseringen kunt maken vanuit je browser, met een eenvoudige grafische interface. Daarnaast heeft het volledige ondersteuning voor Javascript ingebouwd, zodat je in zogenaamde 'function nodes' eigen krachtige logica kunt programmeren, indien gewenst.



Node-RED is ontworpen voor Internet of Things (IOT)-toepassingen en is vooral populair op het gebied van home automation/domotica, vandaar dat dit topic in het forum 'Duurzame Energie & Domotica' staat. Mocht je echter Node-RED-vragen over andere onderwerpen hebben, dan is dat geen enkel probleem in dit topic. Je kunt Node-RED ook standalone gebruiken in combinatie met een 'dashboard'-module.

### Node-RED in combinatie met home-automation apps

Node-RED wordt veel gebruikt in combinatie met applicaties als Home Assistant, Domoticz en OpenHAB. Deze applicaties geven dan veranderingen in de status van je apparaten door aan Node-RED, die alle automatiseringslogica voor je afhandelt en daarna weer aan deze applicaties vraagt om acties uit te voeren, bijvoorbeeld je lampen inschakelen.

Hoewel deze applicaties zelf ook ingebouwde mogelijkheden hebben voor het maken van automatiseringen, is dit niet voor iedereen eenvoudig te begrijpen. Zo krijgt niet iedereen de YAML-taal van Home Assistant, of de LUA-taal van

Domoticz snel onder de knie. Domoticz heeft wel het eenvoudigere 'Blockly', maar dit is minder krachtig. Node-RED lijkt wel een beetje op Blockly, maar is prettiger in gebruik en biedt bovendien meer dan duizend extensions, die 'modules' worden genoemd.

Je moet dit als volgt interpreteren: een sequence (van aan elkaar gekoppelde nodes) wordt geactiveerd, bijvoorbeeld op een bepaald tijdstip, of omdat er een andere trigger is (bijvoorbeeld omdat er iemand thuiskomt, er een deur open gaat, er beweging wordt gedetecteerd, de tv wordt aanzet, etc).

Er gaat dan een berichtje (msg) met een bepaalde waarde (payload) door de sequence heen. Bij elke node kun je het proces laten stoppen, splitsen, of in een andere richting sturen, afhankelijk van de waarde.

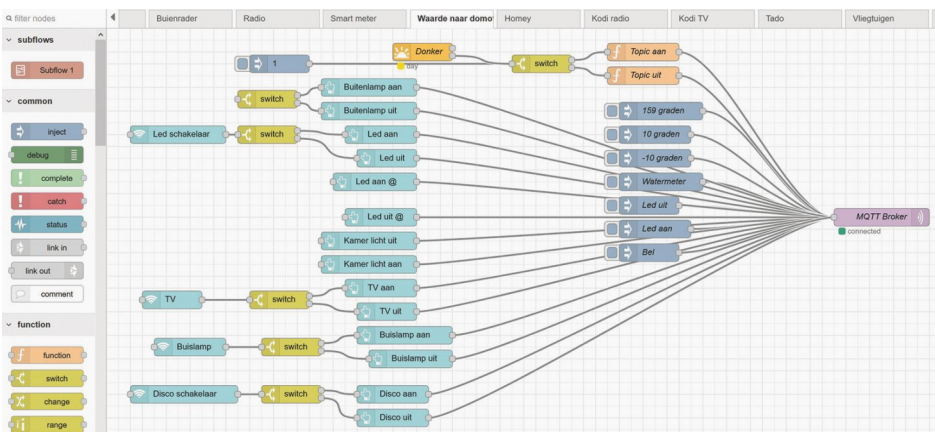
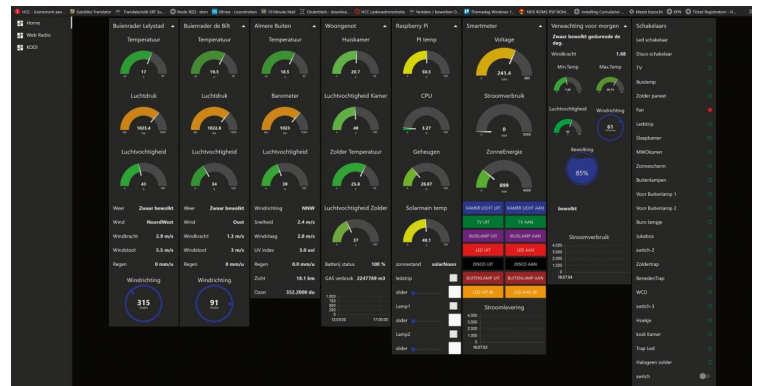
En daarmee kun je verschillend gedrag laten uitvoeren.

### Elkaar helpen bij de HCC!

Als je hulp nodig hebt, dan kun je een screenshot posten, maar het kan ook handig zijn om je 'code' (die Node-RED genereert en die je zeker niet moet kunnen lezen) te delen. Hiervoor selecteer je de relevante nodes in Node-RED, klik je op het hamburgermenu rechts bovenaan en kies je voor *Export* -> *Copy to clipboard*. Die code plak je in je bericht in dit topic in een [code]-blok. En dan ziet het er zo uit:

```
[{"id":"562f35b0.edf0f4","type":"inject","z":"746750be.ec2a","name":"","topic":"","payload":"","payloadType":"date","repeat":"","crontab":"","once":false,"onceDelay":0.1,"x":450,"y":500,"wires":[["e50fb59d.71ff2"]]}, {"id":"e50fb59d.71ff2","type":"debug","z":"746750be.ec2a","name":"","active":true,"tosidebar":true,"console":false,"tostatus":false,"complete":false,"x":620,"y":500,"wires":[]}]
```

Omgekeerd kun je dit soort code ook importeren in Node-RED via datzelfde menu -> *Import* -> *Clipboard* waarna het weer als grafische elementen op het scherm verschijnt.



### De praktijk: HCC-training/workshops

Geïnteresseerd in domotica-zaken, wordt dan lid van de HCC Domotica-IOT interessegroep.

Ga naar: <https://domotica-iot.hcc.nl> meld je aan of abonneer je op de nieuwsbrief voor handige tips en workshops.