Pepper for Dummies

Een handleiding om te leren interageren met Pepper de robot

Inhoudsopgave

[Set-up 3](#_Toc529874595)

[Converseren met Pepper 4](#_Toc529874596)

[Basis 4](#_Toc529874597)

[Een eerste gesprek 4](#_Toc529874598)

[Geneste dialoog 5](#_Toc529874599)

[Stemcommando’s 6](#_Toc529874600)

[Vereisten 6](#_Toc529874601)

[Om te beginnen 6](#_Toc529874602)

[Een eerste commando 6](#_Toc529874603)

[Tips, tricks & troubleshooting 9](#_Toc529874604)

[Help, Pepper reageert niet meer! 9](#_Toc529874605)

[Help, er is geen geluid meer! 9](#_Toc529874606)

# Set-up

[TODO: wireless]

Om een bekabelde verbinding te maken met Pepper volstaat het om Pepper met je computer te verbinden middels een Ethernetkabel.

Start Pepper op via de knop op zijn buik (achter de tablet) en lanceer de Choreographe applicatie op je computer. Ga via het menu naar “Connection” -> “Connect to...”. Als Pepper correct is aangesloten verschijnt hij hier in de lijst met een groen icoontje onder “Status”. Klik op Pepper en vervolgens op de knop “Select”. Na enkele seconden zou de verbinding gelukt moeten zijn, en verschijnt een 3D-model van Pepper in de “Robot view” van Choreographe.

# Converseren met Pepper

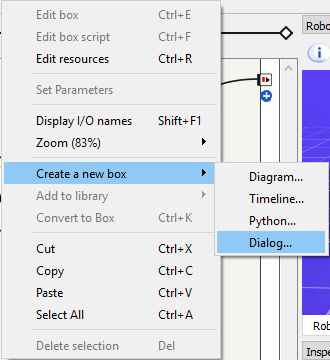
## Basis

Om een gesprek te kunnen voeren met Pepper volstaat het om volgende stappen uit te voeren binnen Choreographe.

Als eerste, voeg een Set Language blok toe en dubbelklik hierop om de taal in te stellen op English:

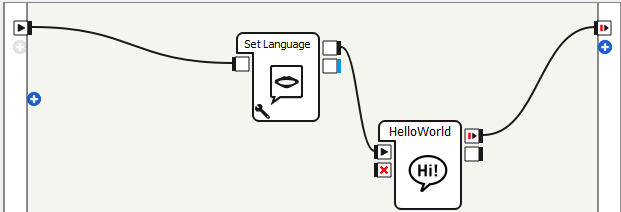


Vervolgens, rechterklik in het werkveld -> Create a new box -> Dialog...:



In het venster dat nu verschijnt, onder Properties, kies “Add Topic”, vul een naam in, en duid de taal aan. Klik vervolgens op “Add”, vul onder “General description” de naam in, en klik op “OK”.

Verbind de 2 blokken met de onStart & onStopped functies zoals hier weergegeven:



## Een eerste gesprek

Nu dat in orde is, kunnen we het gesprek gaan definiëren. Dubbelklik links bij je project files op het \_enu.top bestand zodat er een teksteditor verschijnt.

Pepper kan zelf geen dialoog beginnen, maar hij is wel uitstekend in antwoorden. Typ bijvoorbeeld volgende tekst op een nieuwe regel:

u:(hello) Hello, what is your name?

Uitleg:

* “u:” -> dit duid aan dat we een nieuw gesprek beginnen
* “(hello)” -> indien we dit zeggen, zal Pepper erop reageren
* “Hello, what is your name?” -> dit is hoe Pepper reageert op onze “hello”

Nu gaan we een stapje verder: we gaan leren met variabelen te werken. Op de eerstevolgende regel, typ dit:

u:(my name is \_\*) Nice to meet you $name=$1

Uitleg:

* “\_” -> de volgende input wordt doogegeven aan Pepper
* “\*” -> de gebruiker kan om het even wat zeggen
* “$name=$1” -> hiermee slaat Pepper de eerste input op in een variabele genaamd “name”, zodat hij deze later opnieuw kan gebruiken

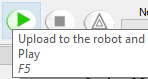
Ten slotte gaan we dit voorbeeld afronden door Pepper onze naam te laten zeggen. Typ volgende tekst:

u:(what is my name) Your name is $name

Uitleg:

* Via $name roepen we de variabele op die we in de vorige stap hebben opgeslagen.

Om het programma naar Pepper te sturen, klik in het lint op de groene “Play”-knop:



## Geneste dialoog

We kunnen alles nog een beetje complexer maken via *nested dialog*, wat wil zeggen: dialoog die mekaar opvolgt, zoals in een echt gesprek. Typ deze regels:

u:(hello) Hello! Are you a student?

u1:(yes) Good luck with your studies!

u1:(no) Are you a teacher?

u2:(yes) thank you for doing such a great job!

u2:(no) Welcome to Odisee!

Uitleg:

* De eerste regel komt bekend voor; we begroeten Pepper en hij stelt ons een vraag
* “u1:(yes)” en “u1:(no)” -> u1 zorgt ervoor dat de “yes” en “no” van de gebruiker enkel aanvaard worden als antwoord op Peppe’s vorige dialoog. Een gebruiker kan dus niet op het begin “yes” of “no” zeggen als hij nog geen “hello” heeft gezegd
* “u2:(yes)” en “u2:(no)” -> hier gaan we nog een niveau verder: we willen nu dat “yes” of “no” enkel aanvaard worden nadat Pepper ons heeft gevraagd of we een docent zijn.

# Stemcommando’s

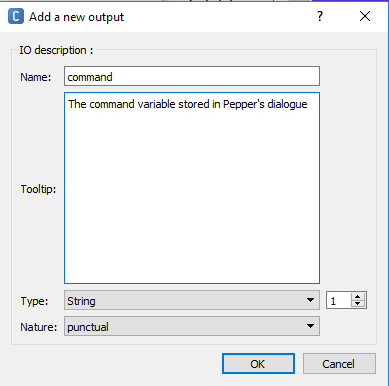
## Vereisten

* Je begrijpt het deel “Converseren met Pepper”, op z’n minst tot en met “Een eerste gesprek”.
* Pepper staat op een plek waar hij voldoende ruimte heeft om zich te verplaatsen

## Om te beginnen

Maak een dialoog aan volgens de stappen beschreven in “Converseren met Pepper” - “Basis”.

Rechterklik de aangemaakte Dialog box en kies “Edit box”. Klik bij “Inputs/Outputs/Parameters” op het blauwe “+”-icoontje rechts van “Outputs”. Vul het venster dat verschijnt zo in:



Uitleg:

* “name” -> dit is de output waarmee we willen werken
* “tooltip” -> een beschrijving van de output die we definiëren
* “Type: String” -> hiermee zeggen we dat onze output een tekst zal zijn

Klik vervolgens twee keer op “OK” om beide vensters te sluiten.

## Een eerste commando

Dubbelklik in het linker venster op het \_enu.top bestand en voeg volgende regels tekst toe:

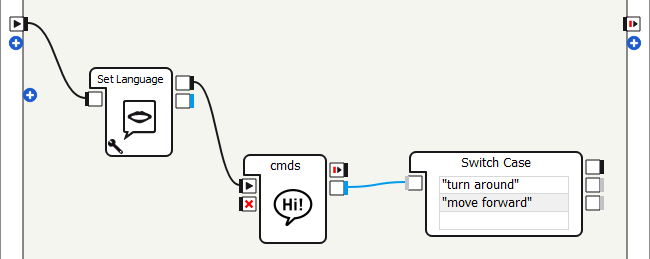
u:(turn around) I will turn around! $command="turn around"

u:(move forward) I will move forward! $command="move forward"

Uitleg:

* Zodra Pepper reageert op één van onze twee stemcommando’s, slaat hij de instructie op in de variabele “command”. Hier gaan we zo meteen mee werken dankzij de output die we in “Om te beginnen” hebben aangemaakt

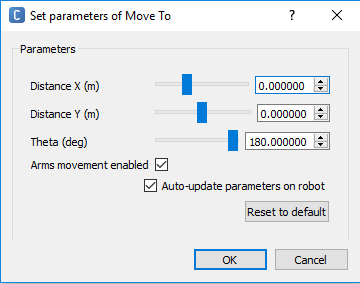
Links bij “Box libraries”, ga naar “Programming” -> “Logic” en voeg een “Switch Case” toe. Typ hierin onze 2 commando’s, elk in een apart vakje, **omringd door aanhalingstekens**. Verbind vervolgens deze Switch Case met het blauwe “command”-blokje van de dialog box. Als het goed is ziet je scherm er nu zo uit:



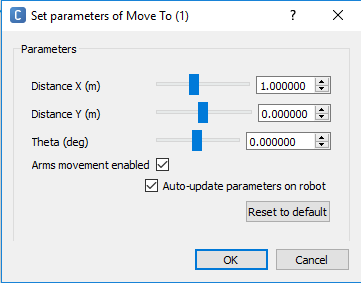
Uitleg:

* Via een Switch Case beperken we het aantal mogelijkheden. Als onze output “command”, die vanuit de dialog wordt doorgegeven, niet één van de twee gedefinieerde waardes heeft, gebeurt er niets.

Nu is het tijd om beweging toe te voegen! Ga via de “Box libraries” naar “Movement” -> “Navigation” en voeg twee keer een “Move To”-blok toe. Klik op het configureer-icoontje (een tang) van het eerste “Move To”-blok. Stel de Distance X & Y in op 0 en stel de Theta in op 180, zoals hieronder weergegeven:



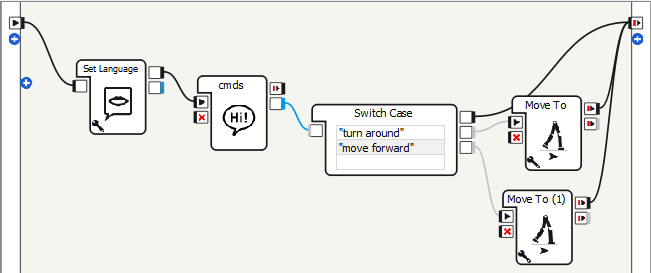
Hiermee geven we Pepper de instructie om 180° rond te draaien in tegenwijzerzin. Klik op “OK” en open de instellingen van het andere “Move To”-blok. Stel hier de “Distance X” in op 1, en de “Distance Y” & “Theta” beiden op 0 zoals hieronder weergegeven:



Verbind nu de blokken op volgende manier:

* “onDefault” van “Switch Case” <-> “onStopped” van Pepper
  + Uitleg: als ons commando iets anders is dan “turn around” of “move forward” moet Pepper geen actie ondernemen
* “output\_1” van “Switch Case” <-> “onStart” van “Move To”
  + Uitleg: “output\_1” komt overeen met “turn around” in de Switch Case
* “output\_2” van “Switch Case” <-> “onStart” van “Move To (1)”
  + Uitleg: “output\_2” komt overeen met “move forward” in de Switch Case
* “onArrivedAtDestination” van “Move To” en “Move To (1)” <-> “onStopped” van Pepper
  + Uitleg: als Pepper klaar is met bewegen, moet hij geen verdere actie uitvoeren

Het eindresultaat zou er zo uit moeten zien:



Nu is het tijd om ons harde werk eindelijk in actie te zien! Upload het programma naar Pepper en zeg eens “turn around”. Als alles goed is verlopen, zal Pepper nu 180° tegenwijzerzin draaien.

# Tips, tricks & troubleshooting

## Help, Pepper reageert niet meer!

Als Pepper niet meer reageert, kan het zijn dat er ergens een fout is opgetreden. In dat geval treedt het zogenaamde “diagnosis effect” in werking:

“Diagnosis effect is a reflex designed to protect the robot and the user in case of a malfunctioning Joint, Actuator or Sensor.”

Als workaround kan je dit effect uitschakelen via de “ALMotion” API. Maak in Choreographe een nieuwe Python box aan. In de functie “onInput\_onStart”, plaats volgende code:

motionService = self.session().service("ALMotion")

motionService.setDiagnosisEffectEnabled(False)

Uitleg:

* Via de eerste lijn code roepen we de “ALMotion” API op
* Via de tweede lijn code laten we Pepper weten dat hij het diagnosis effect moet uitschakelen
* Voor meer info, zie:
  + http://doc.aldebaran.com/2-1/naoqi/motion/reflexes-diagnosis.html
  + http://doc.aldebaran.com/2-1/naoqi/motion/reflexes-diagnosis-api.html

## Help, er is geen geluid meer!

Als er plots geen geluid meer is, kan het zijn dat de output audio gemutet is. Om dat ongedaan te maken, maak een Python box aan en plaats volgende code in de “onInput\_onStart” functie:

audioService = self.session().service("ALAudioDevice")

audioService.muteAudioOut(False)

Uitleg:

* Via de eerste lijn code roepen we de “ALAudioDevice” API op
* Via de tweede lijn code laten we Pepper weten dat de output audio **niet** gemutet mag zijn
* Voor meer info, zie:
  + http://doc.aldebaran.com/2-1/naoqi/audio/alaudiodevice-api.html