

Dialogo con NAO

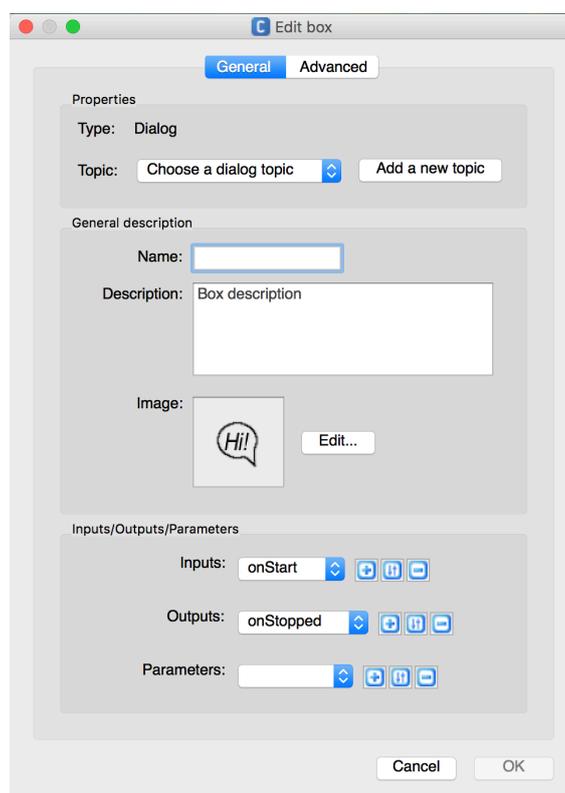
Materiale necessario:

- NAO (e alimentatore)
- Computer connesso alla stessa rete locale del robot

Procedimento:

Choregraphe offre la possibilità di creare box “dialog”.

Tasto destro sul flow diagram panel > *Create a new box > Dialog...*



Click su “Add a new topic”.

Trattandosi di un esempio, il nuovo topic è: dialogo_di_prova. Selezionare la lingua supportata: italiano; togliere il flag dalla lingua inglese.

Infine, click su “Add”. Click sul bottone “OK” per confermare la creazione del nuovo box

Nota bene: il nome del nuovo topic non deve contenere spazi!

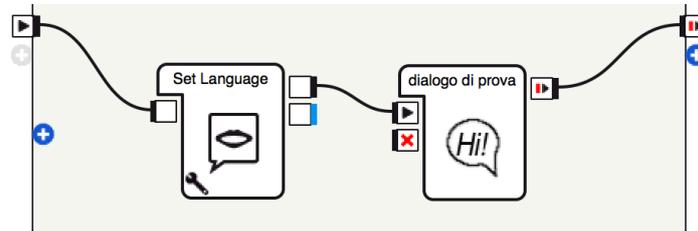
È preferibile utilizzare il carattere underscore _ .



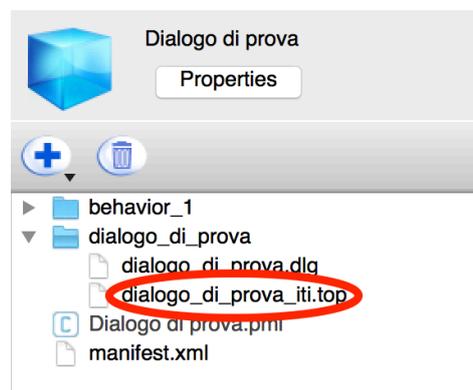
Collegare l’output del box **Set Language** all’input del box creato precedentemente.

Impostare i parametri del box “Set Language” cliccando sulla chiave inglese. Impostare la lingua italiana.

Ecco l’aspetto del programma finale:



Nel pannello Project content, click sul file che termina con estensione `.top`.



Si aprirà un editor di script.

ALDialog

ALDialog è un modulo che permette di dotare il robot di abilità nell’interagire con un umano, sulla base di alcune *rules* scritte nel modo appropriato. Per meglio gestire le conversazioni fra utente e robot, le *rules* sono raggruppate per argomento.

Una *rule* associa ad un input dato dall’umano (ciò che dice) ad un output del robot (ciò che il robot risponde).

Primo script:

```
topic: -dialogo_di_prova()  
language: iti
```

```
u:(Ciao NAO) Ciao, questa funzionalità è davvero forte.
```

La prima riga di codice specifica il topic delle interazioni. La seconda la lingua. Fra parentesi tonde è contenuto l’*human input*, mentre dopo le parentesi segue l’output pronunciato dal robot.

Rendiamo la *rule* un po’ più complessa:

```
topic: -dialogo_di_prova()  
language: iti
```

```
u:(Ciao [NAO robot]) Ciao, questa funzionalità è davvero forte.
```

Le parentesi quadre rendono la rule più flessibile, in questo modo accetta alcune variazioni. Nel caso la variazione desiderata sia di una frase intera, la frase deve essere contenuta fra virgolette " ... ".

User subrule

```
topic: -dialogo_di_prova()
language: iti
```

```
u:(Ciao [NAO robot]) Ciao, questa funzionalità è davvero forte.
u:(Facciamo {"un po' di"} attività fisica) Volentieri, piegamenti o tai chi?
  u1:(Piegamenti) Teniamoci in forma!
  u1:(tai chi) Vada per il tai chi!
```

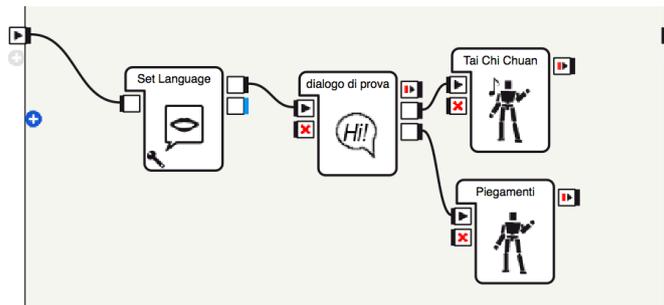
Le parentesi graffe contengono una parola o una frase opzionale. Si possono trovare sia in un *human input* sia nel *robot output*.

Le *subrule* permettono di creare alcuni contesti di conversazione, attivandosi se e solo se una rule principale è stata attivata via *human input*.

Collegare Qichat script e animazioni

Al box dialog creato in precedenza, aggiungere due output:

- TaiChi - tipo "bang"
- Piegamenti - tipo "bang"



```
topic: -dialogo_di_prova()
language: iti
```

```
u:(Ciao [NAO robot]) Ciao, questa funzionalità è davvero forte.
u:(Facciamo {"un po' di"} attività fisica) Volentieri, piegamenti o tai chi?
  u1:(piegamenti) Teniamoci in forma! $Piegamenti=1
  u1:(tai chi) Vada per tai chi! $TaiChi=1
```

Le funzionalità del carattere dollaro \$ sono svariate, in questo caso, se il dollaro è seguito dal nome dell'evento output **E** ad esso viene assegnato 1, verrà stimolato l'output relativo.

Testare i programmi

1. **Robot Virtuale:** Per simulare la pronuncia delle parole su un robot virtuale, visualizzare il pannello "Dialog" facendo click su *View > Dialog*. Basterà inserire nel campo di testo la frase desiderata e premere invio.
2. **Robot reale:** L'impostazione *Autonomous life* deve essere attiva. Premere play, verrà riprodotto un "bip" e gli occhi diventeranno blu: sono i segnali che il robot sta ascoltando. Parlare velocemente.

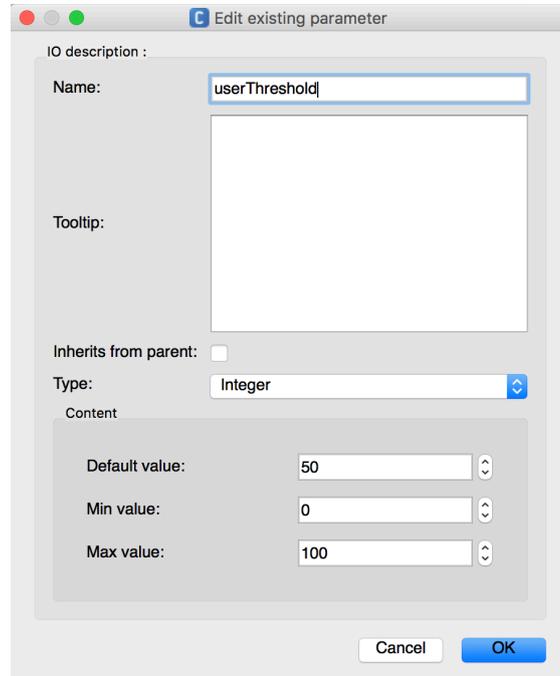
Per maggiori dettagli si consiglia la lettura della documentazione:

http://doc.aldebaran.com/2-1/naoqi/audio/dialog/dialog-syntax_full.html

Modificare la *confidence threshold*

In alcuni casi può essere utile modificare la soglia con cui viene validato il risultato prodotto dal motore di riconoscimento vocale automatico (ASR engine).

Creare un nuovo box python e aggiungere ad esso un parametro di tipo "integer" nominandolo: *userThreshold*. Default value: 50.



Codice contenuto nel box:

```
class MyClass(GeneratedClass):
    def __init__(self):
        GeneratedClass.__init__(self)

    def onLoad(self):
        pass

    def onUnload(self):
        pass

    def onInput_onStart(self):
        ALDialog = ALProxy("ALDialog")

        t = self.getParameter("userThreshold") /100.
        ALDialog.setASRConfidenceThreshold(t)

        t = int(t*100)
        self.log("\n\tNew threshold is: %d%%" %t)
        self.onStopped()

    def onInput_onStop(self):
        self.onUnload()
        self.onStopped()
```

Flow diagram panel:

