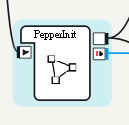
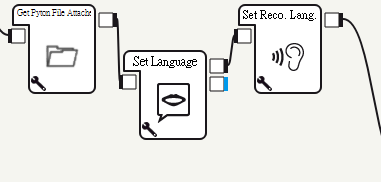
APP流程方面



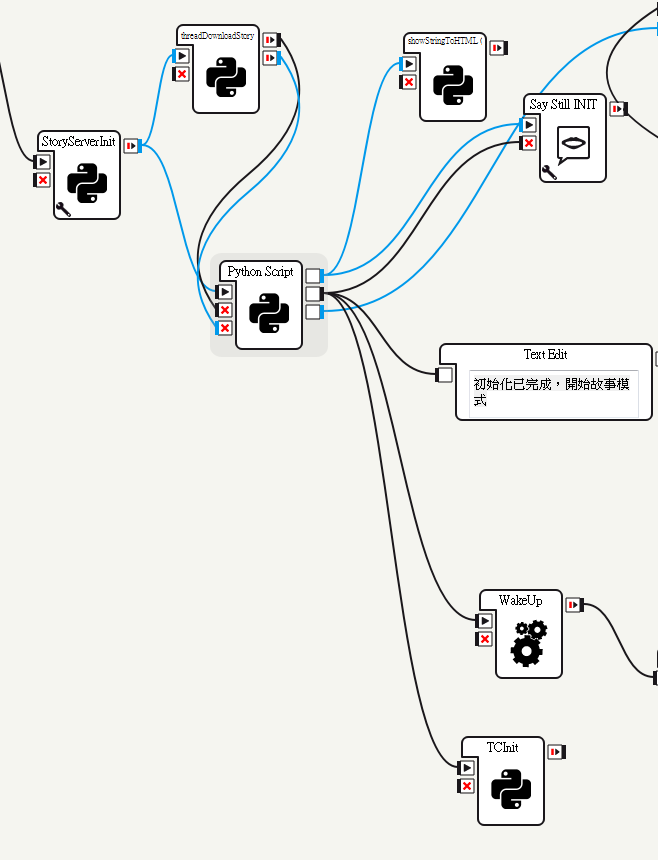
In PepperInit



設定 1. Py檔位置，未來可以在其他block使用

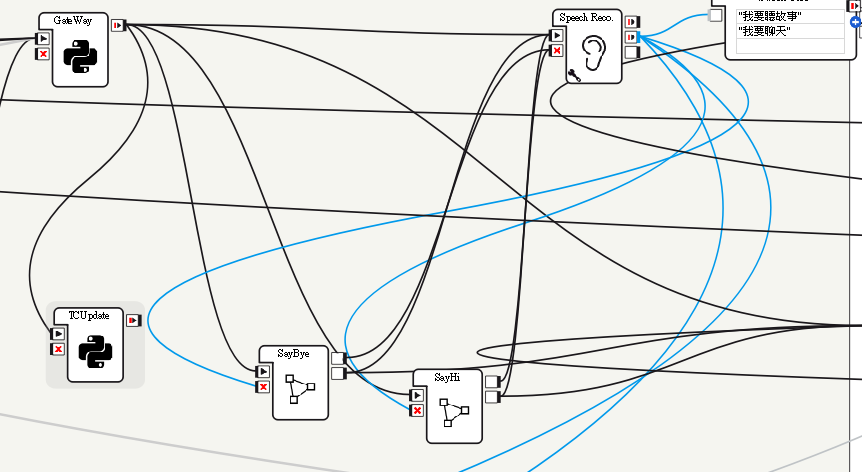
2.TTS語言設定

3.辨識關鍵詞語言設定



在StoryServerInit 是去FTP Server 下載名為init.txt檔案裡的故事清單，並暫存在Memory裡面，並透過ThreadDownloadStory分五個Thread下載故事檔，在下載同時，機器會看local端 故事檔數量來知道故事下載完成了百分比，並隨之撥放，直到全部下載完後才進入至初始化最終階段，Wakeup機器及TCInit，TCInit為下載機器人所念的TTS的內容，如 今天要聽什麼故事 等等可變動語言

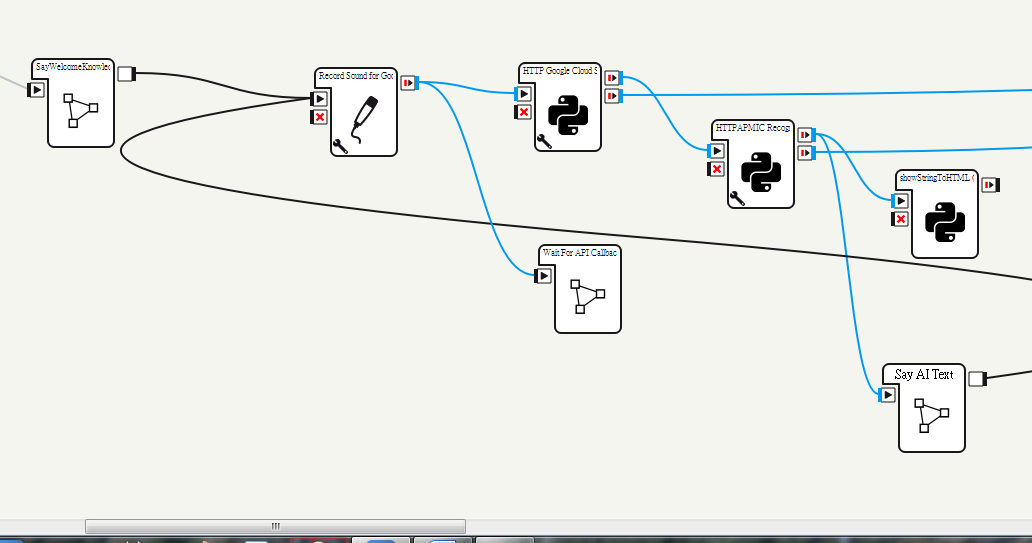
初始化完後



SayBye 及 SayHi 為當機器人尚未接受到指令時，可透過碰觸頭跟手來與她互動，碰頭會進入至SayHi，碰手會進入至SayBye

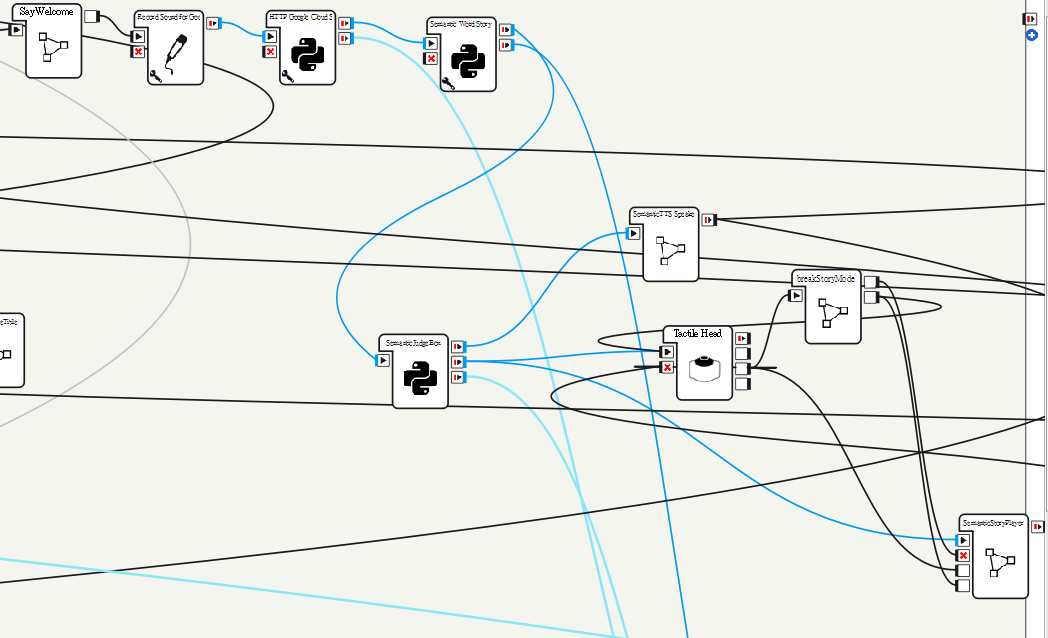
如果當Speech Reco辦別出 “我要聽故事”或”我要聊天”時，進入至下一階段。

我要聊天:



此階段依樣會先講歡迎語”如你今天要知道什麼事呢?”，之後錄音5秒在上傳給Google Cloud Speech API去取得此人語音的文字，並傳給亞太智能引擎API，透過此Sessem 可以有多段對話。

我要聽故事

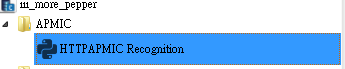


一樣透過Google STT API來知道使用者語言文字，並傳給Semantic Word Story Box來去知道他講了什麼內容，再透過SemanticJudgeBox來知道 Server是要回答TTS還是故事名稱

1. 如果是TTS部分(像是”我不清楚你在講什麼”) 說完就回初始狀態，繼續等關鍵詞
2. 如果是故事名稱，會先去Local端對應故事名稱並撥放，同時會開啟頭的偵測，如果再撥故事期間，頭被Touch時，會中斷故事，並詢問是否有什麼問題，如果是不想聽了，就回初始狀態，繼續等關鍵詞。如果是誤觸，則繼續撥放。

SDK部分

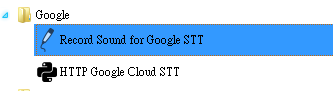
1.

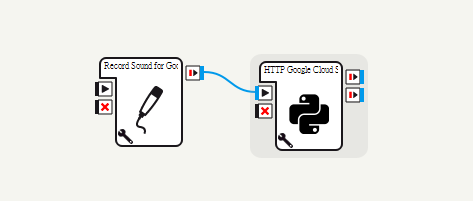


Input : String: 想問的問題文字內容

Output String: 解答的文字內容

2.





Input Signal

Output: 語音轉文字的文字內容，如果無文字會在OnFailed出去

3.



Input 預想聽的故事文字

Output: 此故事名稱或TTS，如果連線有問題會在OnFailed出去

4.



Input Signal

Output Signal

此為找尋全部在Project 根目錄底下py檔做連結

5.

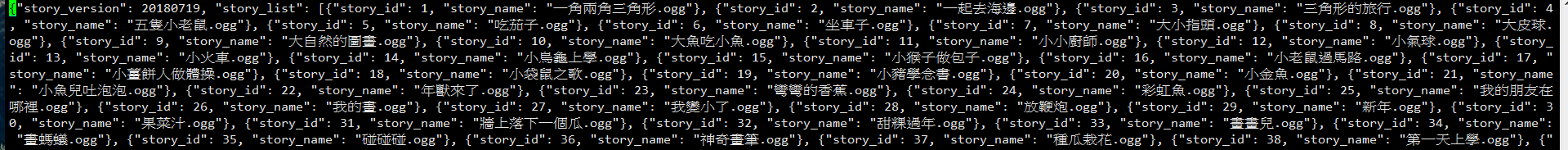


Input Signal

Output json Text

此為去FTP撈取最新的故事 init檔，並判斷local端檔案版本(story\_version)是否跟Server一樣，不一樣的話，會全部將local端故事刪除並重新下載，如果一樣會檢查比對local端故事是否缺少故事，並重新下載

以下為init.txt檔案內容



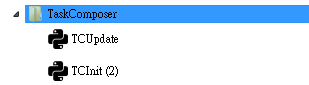
6.



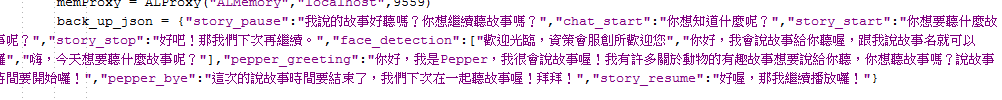
Input 欲在面板展示文字

Output Signal

7.



TCUpdate透過HTTP協定並在Memory更新說話內容，如以下表



TCInit異同TCUpdate