UNIVERSITÀ DEGLI STUDI DELL'AQUILA

Dipartimento di Ingegneria e Scienze dell'Informazione e Matematica

Università degli Studi dell'Aquila

Cognitive Robotics Systems Engineering

A PILOT PROJECT OF AN EMPATHIC ROBOT

Università degli Studi dell'Aquila - DISIM - Ingegneria Informatica e Automatica

Supervisor: Professor Giovanni De Gasperis

Co-Supervisor: Professor Laura Tarantino

Student: Agnese Salutari

BlocksBot

Many different robots working with people:

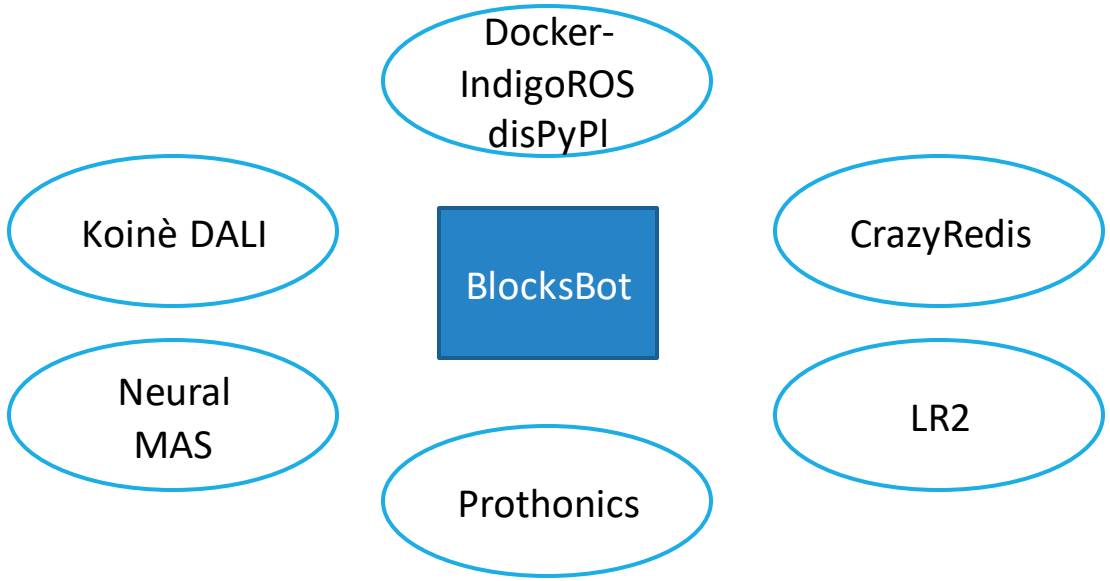
BlocksBot solution to easily design and program them all

What is BlocksBot?

BlocksBot is a pilot project of an Empathic Robot:

- A Robot is Empathic when it can recognize human emotions and properly reacts
- Reaction can comprise some action and the simulation of an emotion
- Empathic Robotics is now spreading fast

BlocksBot works for both real world and simulation robots.



Why BlocksBot?

BlocksBot provides common and easy to use tools
for building Empathic Robots as hybrid distributed systems
in simulations and real world. It is:

- Portable
- Modular
- General
- Highly compatible

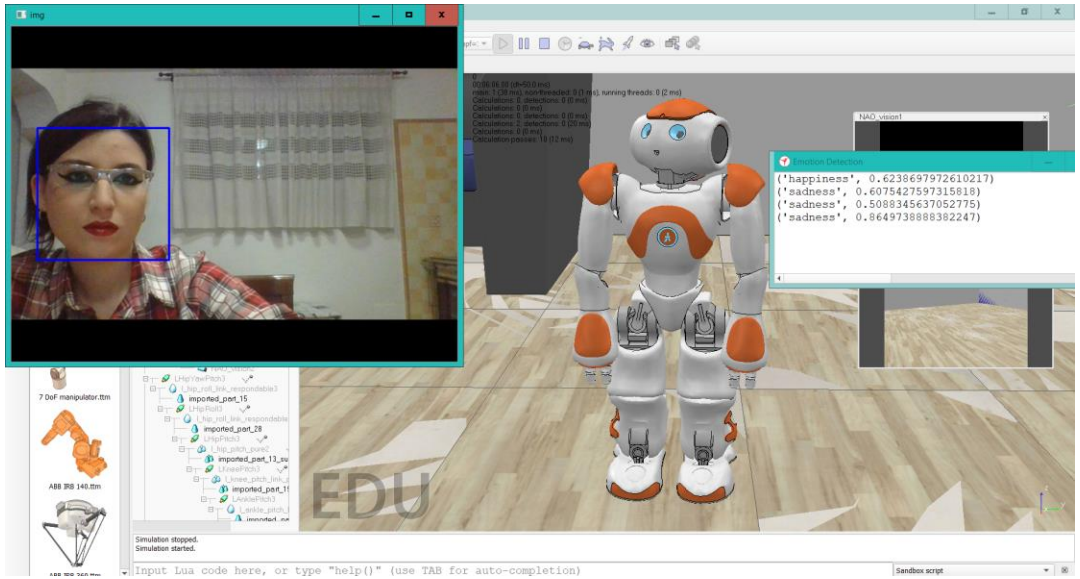
What can a BlocksBot Robot do?

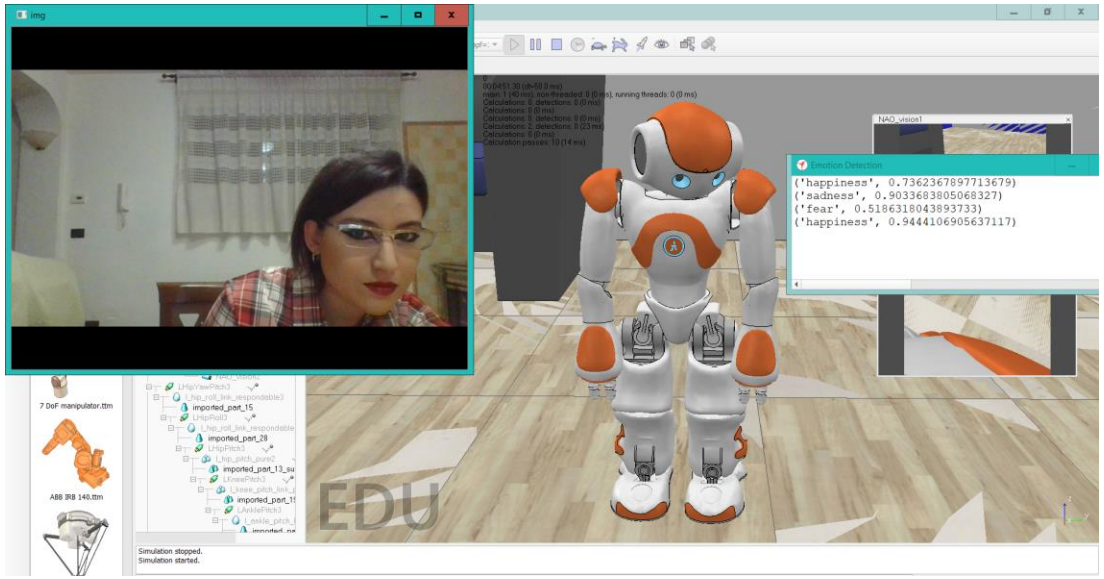
Robot detects emotions from:

- Facial expressions
- Pose
- Voice

Combines them getting the most probable one.

Then reacts as configured.





BlocksBot in action

Video

Hybrid Approach

Procedural Programming:

- Specifying the process step by step

+

Logic Programming:

- Describing the start state, the wanted state and the rules that can determine the change between adjacent states

+

Machine Learning (for example neural networks):

- Creating the module and training it

python

SWI Prolog

SICStus

Keras

TensorFlow

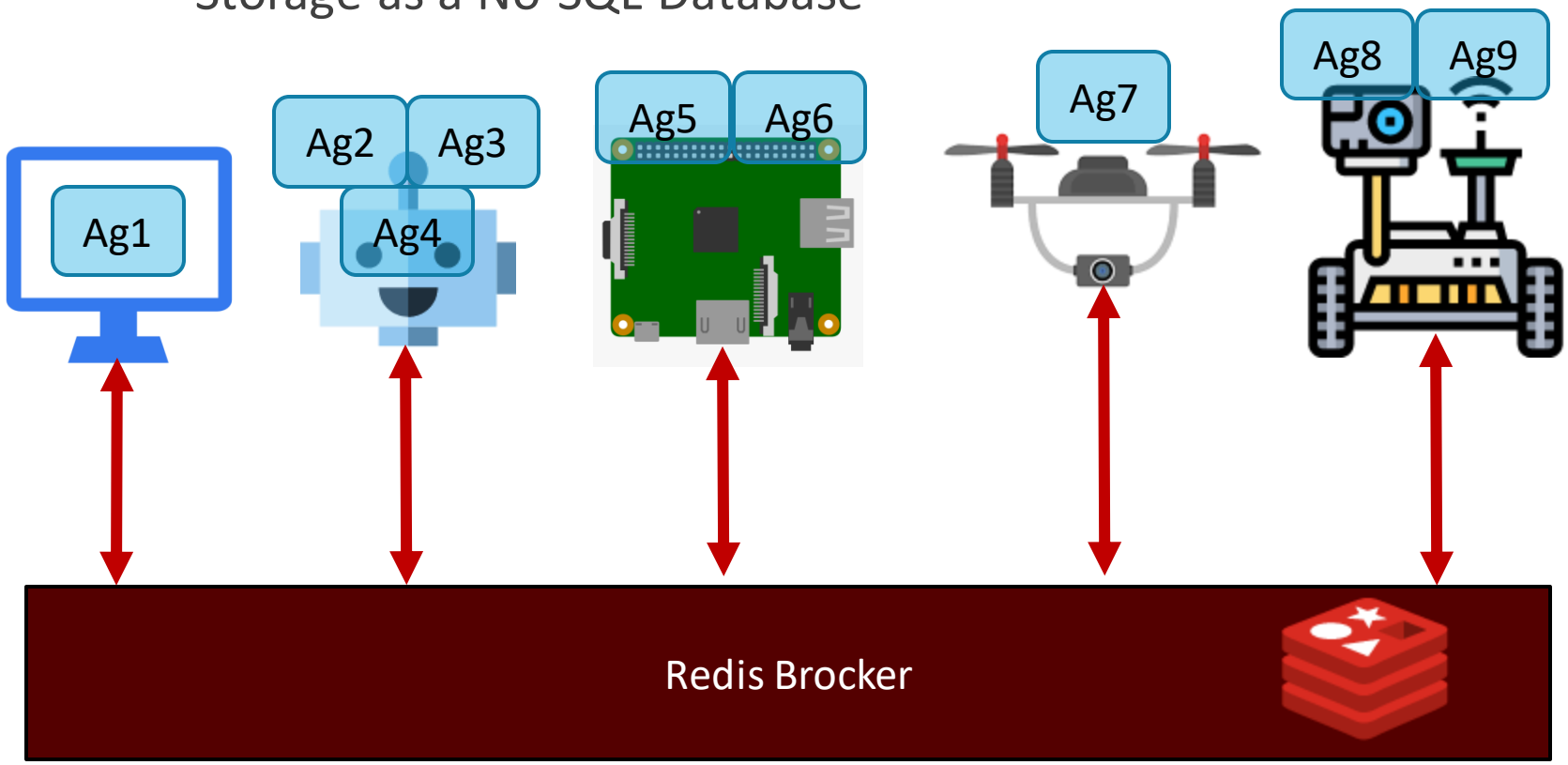
Distributed Approach

BlocksBot is a Distributed Hybrid System:

- It is a set of processes, agents, running on different machines
- Each agent can be based on a different technology

Redis is used as a brocker:

- Publish/Subscribe on Channels
- Storage as a No-SQL Database



Technologies for Emotion Detection

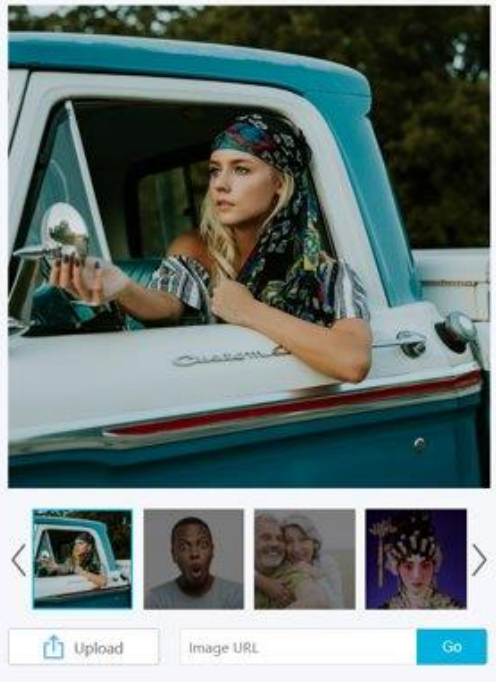
Face++:

- Face and facial emotion detection (anger, disgust, fear, happiness, neutral, sadness, surprise)
- Body and posture detection

Vokaturi:

- Vocal emotion detection (neutrality, happiness, sadness, anger, fear)

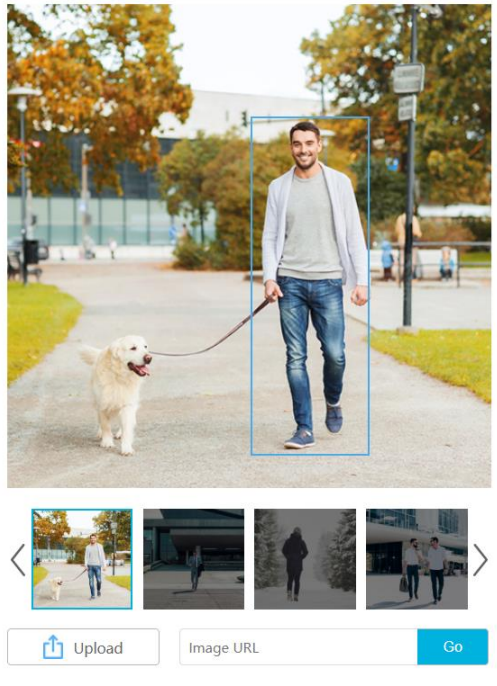
Face++ face detection



Results | Response JSON


```
{
 "request_id": "1583599891,7dfa3a4c-c000-42cd-a54c-e374767ffb77",
 "time_used": 692,
 "faces": [
  {
   "face_token": "5c07813ff195188114a5d7451029c49c",
   "face_rectangle": {
    "top": 379,
    "left": 506,
    "width": 162,
    "height": 162
   },
   "attributes": {
    "emotion": {
     "anger": 0.004,
     "disgust": 0.004,
     "fear": 0.004,
     "happiness": 0.363,
     "neutral": 75.48,
     "sadness": 0.378,
     "surprise": 23.766
    }
   }
  }
 ],
```


Face++ body detection



Results | Response JSON


```
{
 "image_id": "g4+pvOcb2Rc/l9RBmTstPw==",
 "request_id": "1583599976,30a29a92-9545-4001-8d66-5b900e53b61e",
 "time_used": 759,
 "skeletons": [
  {
   "body_rectangle": {
    "width": 119,
    "top": 118,
    "left": 253,
    "height": 347
   },
   "landmark": {
    "right_hand": {
     "y": 163,
     "x": 17,
     "score": 0.51721567
    },
    "head": {
     "y": 4,
     "x": 54,
     "score": 0.49851495
    },
    "neck": {
```

My BlocksBot Agents

(available on GitHub)

Emotion detection and decision:

- FacialEmotionsAgent
- PoseEmotionsAgent
- VocalEmotionsAgent
- DecisionMakerAgent

Images and audio capturing and reacting for simulations:

- AudioSimulationManagerAgent
- VideoSimulationManagerAgent

Images and audio capturing and reacting for real robot:

- AudioNaoBotManager
- VideoNaoBotManager
- ReactionNaoBotManager

Configurable System Starters:

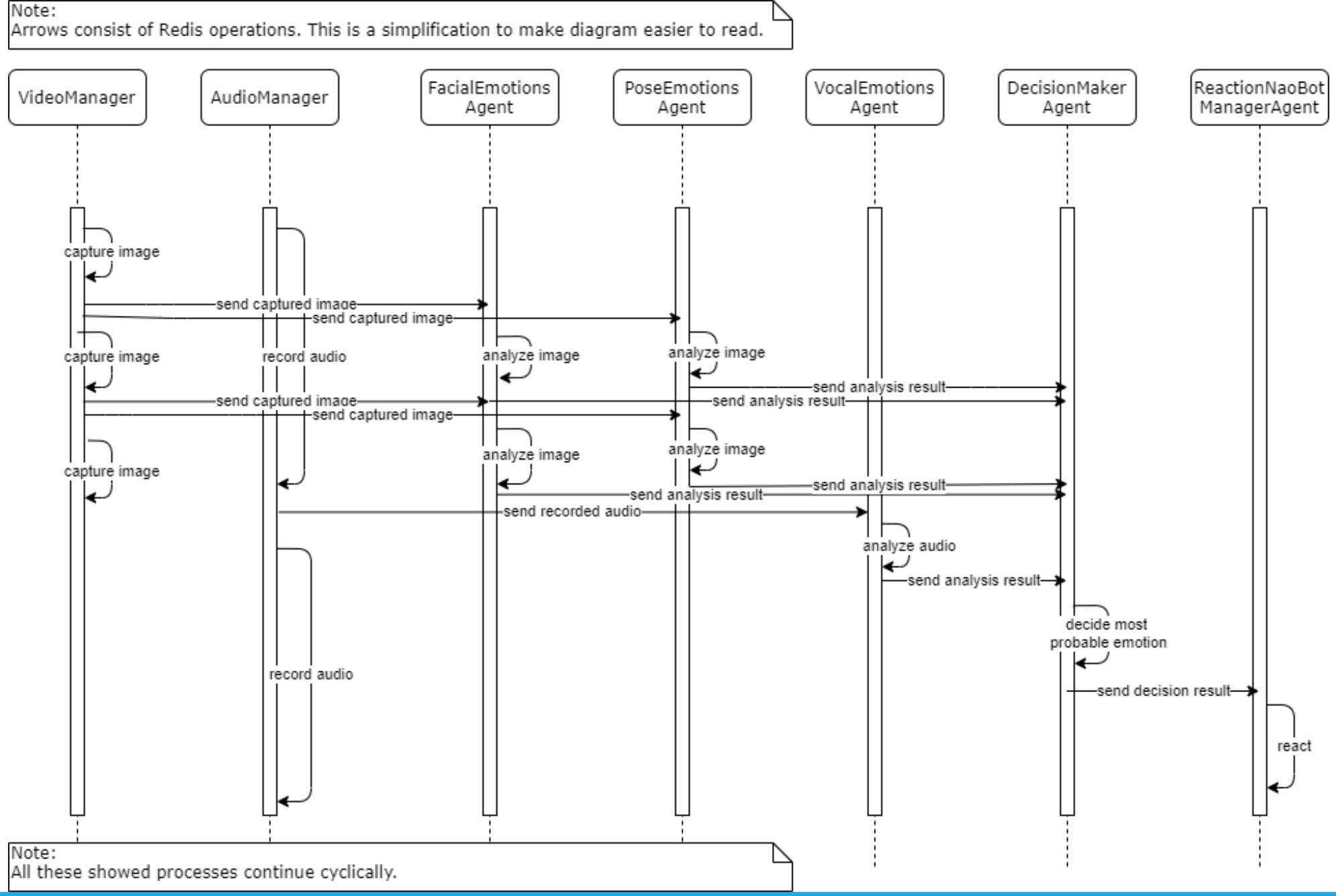
- Python 2 Runner for real Nao
- Python 3 Runner

Face++ 旷视 AI Open Platform

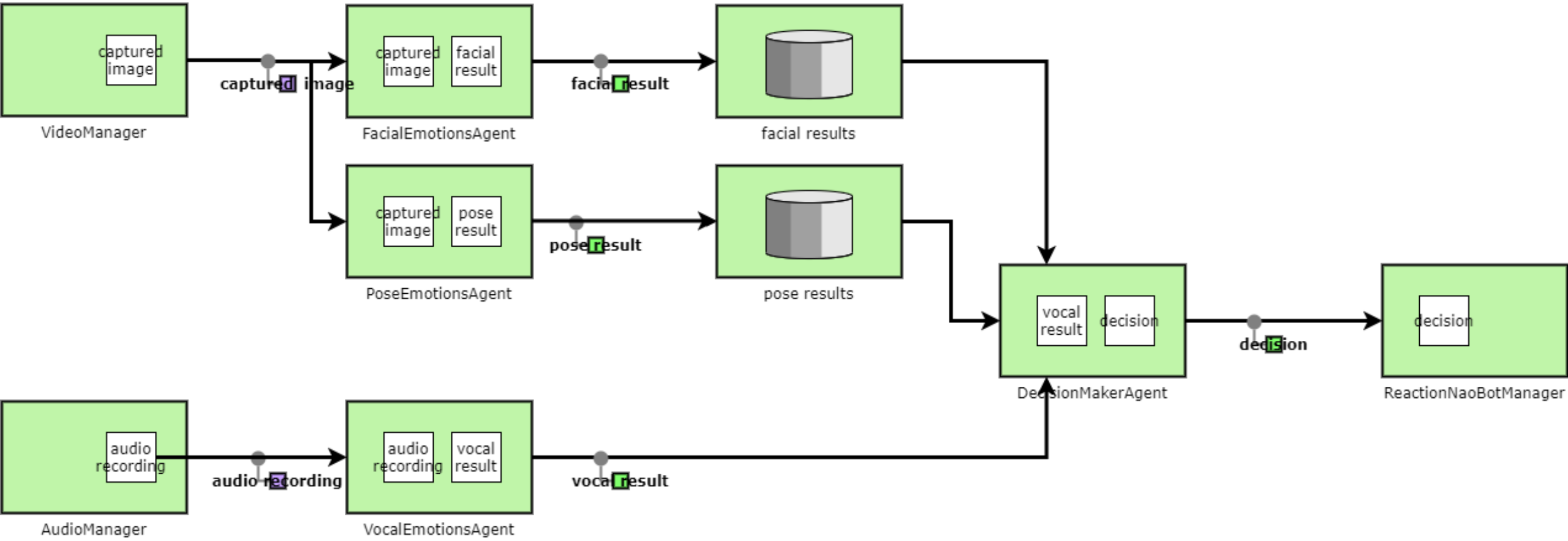
VOKATURI

OpenCV

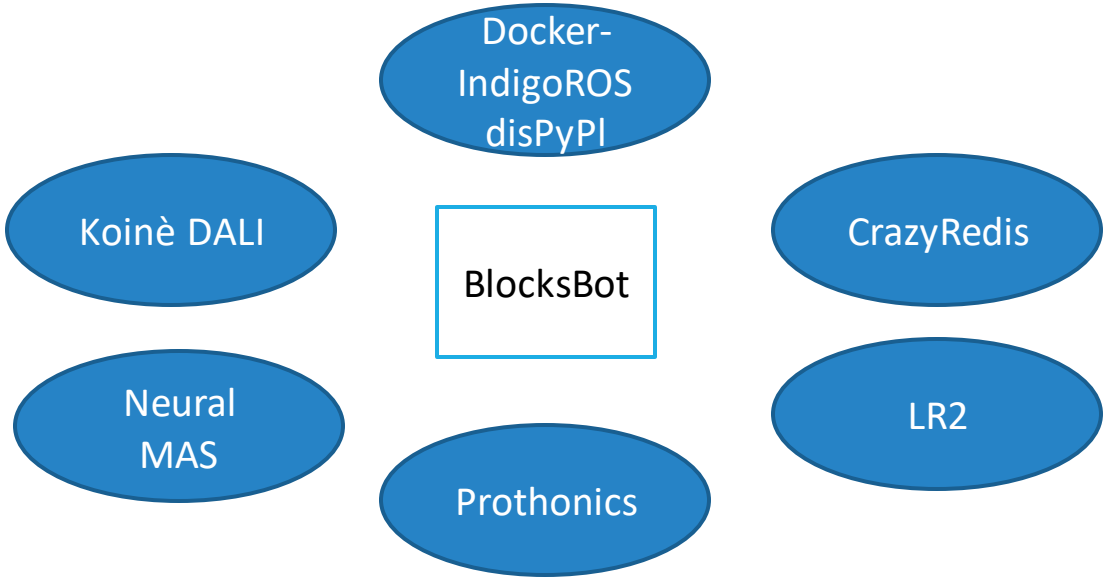
Sequence Diagram



Enterprise Integration Patterns Diagram



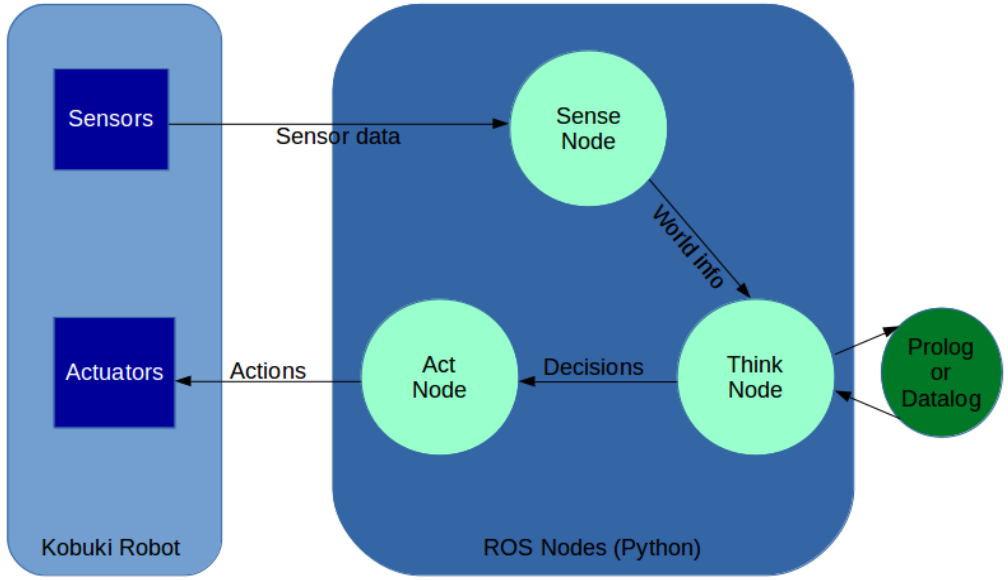
My Pluggable Robotics Projects



(available on GitHub)

Docker-IndigoROSdisPyPl & KobukiROSIndigo

- Docker-IndigoROSdisPyPl is a Docker container with all the needed tools to create Python, Python/SICStus-Prolog, Python/SWI-Prolog and Python/Datalog agents in a ROS environment
- KobukiROSindigo is an example of such a system



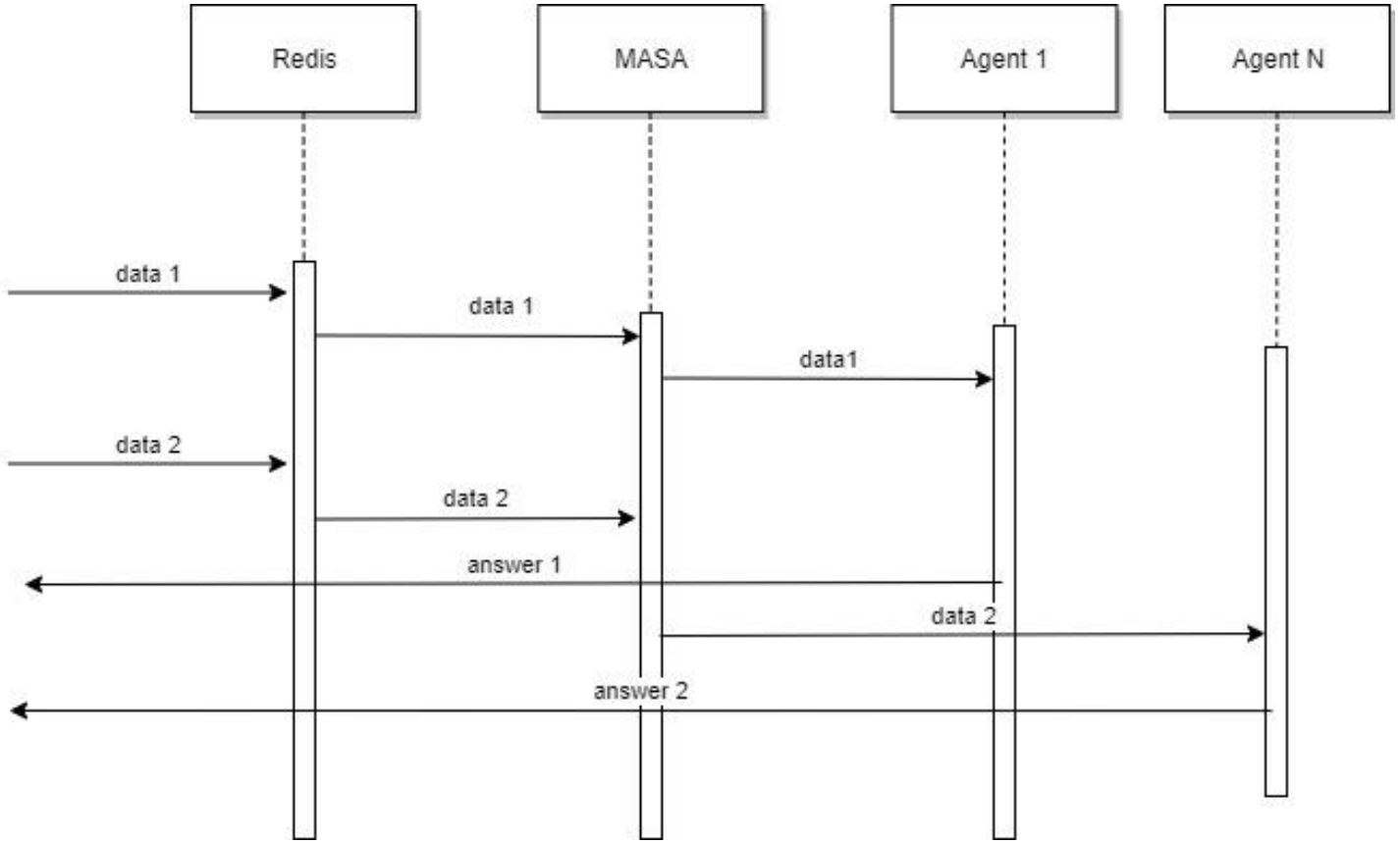
Koinè DALI

My extension of DALI framework.

Distrubuted system with:

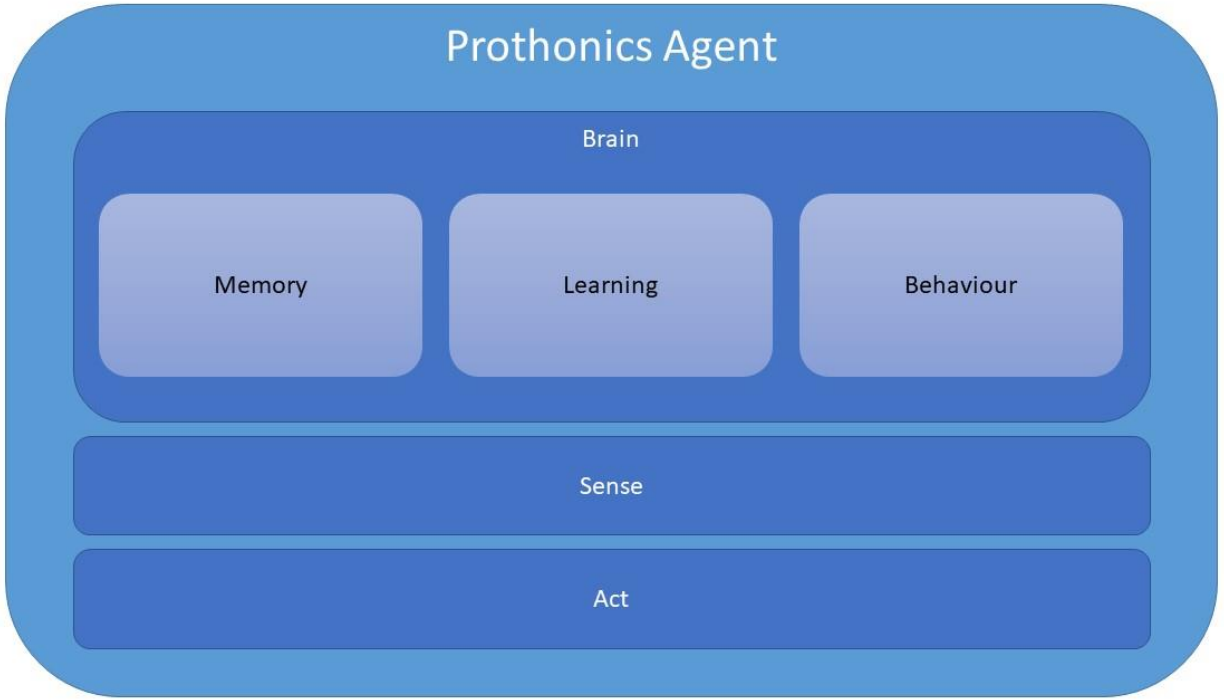
- Different MASs
- Object Oriented modules

connected together via Redis.



Prothonics

Prothonics allows to create agents that can perform reasoning using SWI-Prolog in V-REP or CoppeliaSim simulations.

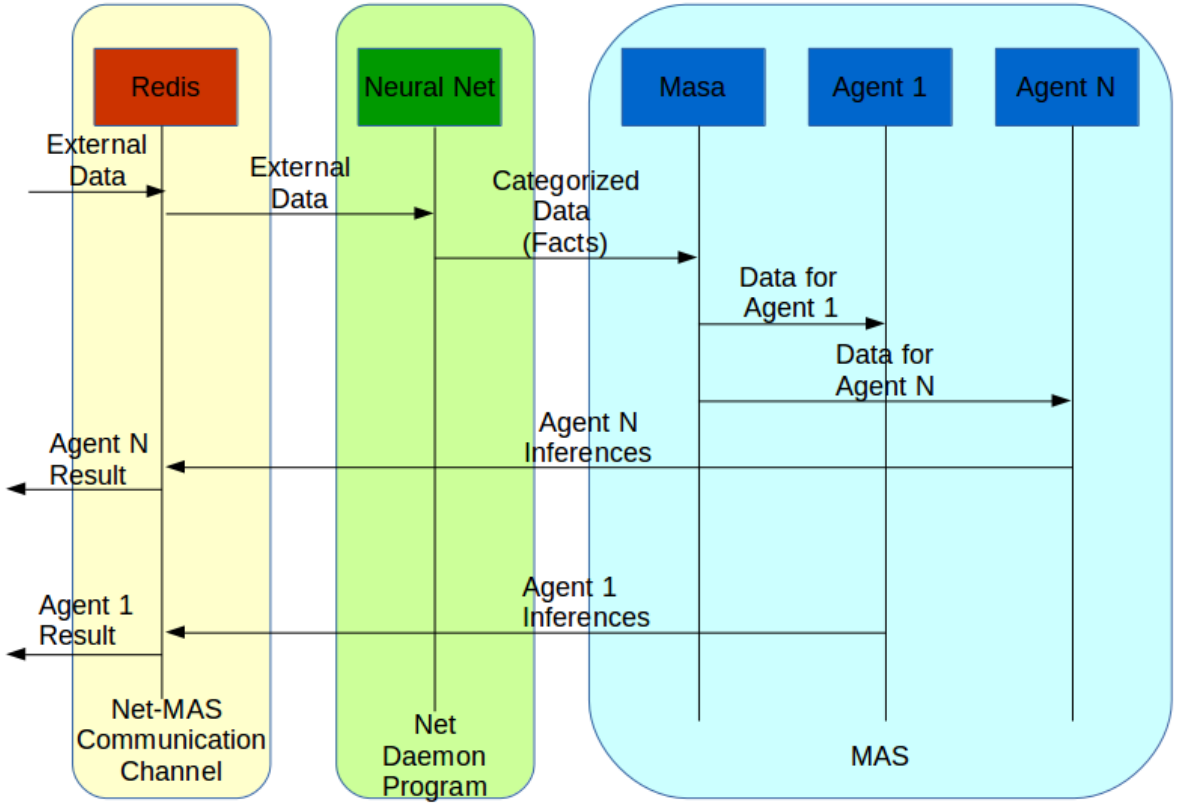


NeuralMAS

It allows to work with both MASs and Neural Networks.

It includes:

- A Keras Neural Net
- A Koinè DALI MAS
- A Redis broker



CrazyRedis

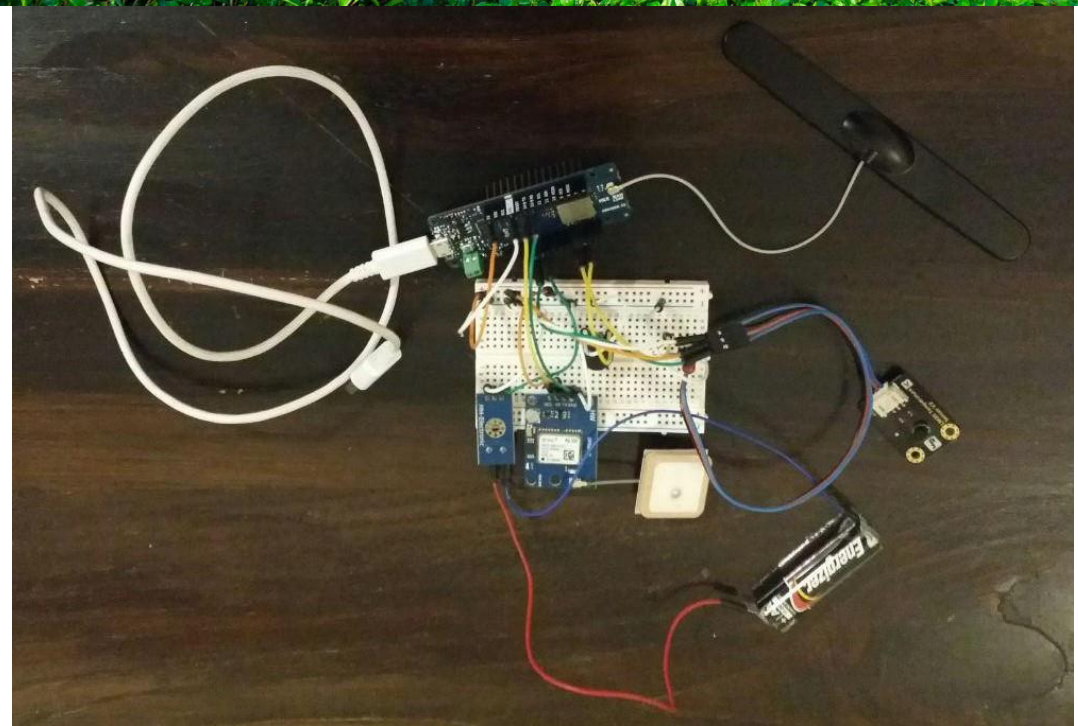
Python 3 library
to easily manage Crazyflie 2 drones API
with callbacks and
to receive log data via Redis.

It can be used by a robot.

Lora Robot Rescue

Hardware/Software module to:

- Periodically send GPS location, battery level and temperature via a LoRaWAN connection
- Eventually send an alert and signal the robot to enter in powersafe mode
- Make the robot more evident, when robot owner is looking for it



Future developments

- Perform more tests on real Nao
- Improve pose emotion detection
- Improve logic
- Expand BlocksBot library
- Work on Python 3 robots

Acknowledgements

Thanks to Università degli Studi dell'Aquila Professors and Staff for having helped me to grow professionally and as a person and for their diligence in these difficult days.
I'd especially like to thank Professor De Gasperis and Professor Tarantino for their support and patience.
Thanks to my parents and family for their closeness and encouragement.
Finally, thank you for your attention.