# Project No. 611391

# DREAM
# Development of Robot-enhanced Therapy for Children with Autism Spectrum Disorders

Grant Agreement Type: Collaborative Project
Grant Agreement Number: 611391

# D6.4 Expression and Actuation Subsystem

Due date: **1/4/2015**
Submission Date: **25/3/2015**

Start date of project: **01/04/2014** Duration: **54 months**

Organisation name of lead contractor for this deliverable: **Vrije Universiteit Brussel**

Responsible Person: **Bram Vanderborght** Revision: **2.2**


| Project co-funded by the European Commission within the Seventh Framework Programme | | |
|---|---|---|
| **Dissemination Level** | | |
| **PU** | Public | **PU** |
| **PP** | Restricted to other programme participants (including the Commission Service) | |
| **RE** | Restricted to a group specified by the consortium (including the Commission Service) | |
| **CO** | Confidential, only for members of the consortium (including the Commission Service) | |

# Contents

## Executive Summary

Deliverable D6.4 defines the specification, design, implementation and validation of the Expression and Actuation subsystem within the cognitive architecture in Work Package 6.

More specifically, this report presents the advances done in task T6.4 for the first year of the DREAM project. During this first year the cognitive architecture and the Expression and Actuation subsystem have been designed. They are described within this report.

## Principal Contributors

The main authors of this deliverable are as follows (in alphabetical order).

Paul Baxter, Plymouth University
Tony Belpaeme, Plymouth University
Hoang-Long Cao, Vrije Universiteit Brussel
Albert de Beir, Vrije Universiteit Brussel
Pablo Gómez, Vrije Universiteit Brussel
Emmanuel Senft, Plymouth University
Greet Van de Perre, Vrije Universiteit Brussel
Bram Vanderborght, Vrije Universiteit Brussel

## Revision History

Version 1.0 (P.G. 27-01-2015)
First draft.

Version 2.0 (P.G. 26-02-2015)
Some texts have been corrected.

Version 2.1 (P.G. 23-03-2015)
Appendices added.

Version 2.2 (P.B. 25-03-2015)
Addition of details of preliminary version of Expression and Actuation subsystem component.

# 1 Overview of WP6 architecture

In DREAM we will move away from Wizard of Oz-controlled behaviour for the robot, which too often is the de facto mode of interaction in Robot Assisted Therapy [1]. Therefore work package WP6 aims to progress the theoretical and methodological understanding of how an embodied system can interact autonomously with young users in a learning task, specifically developed for atypically developing children. WP6 is concerned with the development of the robot behaviour subsystems to provide social robots with a behaviour underlying social interaction, which permits the robot to be used in RET in a supervised autonomous way. This involves both autonomous behaviour and behaviour created in supervised autonomy, whereby an operator requests certain interventions, which are then autonomously executed by the robot.

A general high level description of the robot control system is shown in figure 1. This basically describes how the autonomous controller is informed by three external sources: the child behaviour description, sensory information, current intervention script state, and input from a therapist (e.g. emergency stop, not shown in diagram). Combining these sources, the autonomous controller should trigger as an output the appropriate sequence of action primitives to be performed (as well as some feedback via the WoZ GUI), which then gets executed on the robot.

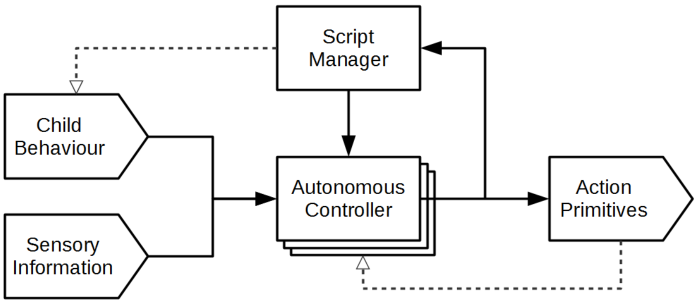


Figure 1: High level description of the robot control system. Child behaviour interpretation (WP5) and sensory information (WP4) provide the context for the autonomous action selection (as well as feedback from motor command execution), in combination with the particular intervention script being applied. The intervention script provides context for child behaviour interpretation.

The autonomous controller is composed of a number of subsystems, as described in the DoW: Reactive, Attention, Deliberative, Self-Monitor and Expression and Actuation. In the Reactive subsystem, sensory inputs are immediately acted upon with appropriate actuator outputs. The Attention subsystem determines the robot's focus of attention. In the Deliberative subsystem, the necessary interventions will be implemented in a general approach so it is not scenario-specific. The Self-Monitoring subsystem acts as an alarm system in two specifications. An internal one when the robot detects that it cannot act because of a technical limitation or an ethical issue. An external alarm is one where the therapist overrules the robot behaviour selection. Finally, the Expression and Actuation subsystem is responsible for generating believable human/animal-like smooth and natural motions and sounds that are platform independent. These subsystems interact, and must combine their suggested courses of actions to produce a coherent robot behaviour, in the context of constraints laid down by the therapist (for example, the script to be followed, types of behaviour not permissible for this particular child because of individual sensitivities, etc). As a result, we have formulated the following architecture describing how cognitive control informed by the therapy scripts is to be achieved (figure 2), see

Annex 3.5 for further details. A detailed description of the cognitive architecture will be provided in deliverable D6.1 at month 18. Next version of this document will be ready for month 24.

Within this report we describe the functionality of the Expression and Actuation subsystem as well as any other additional module required to execute the motor commands in the robot.

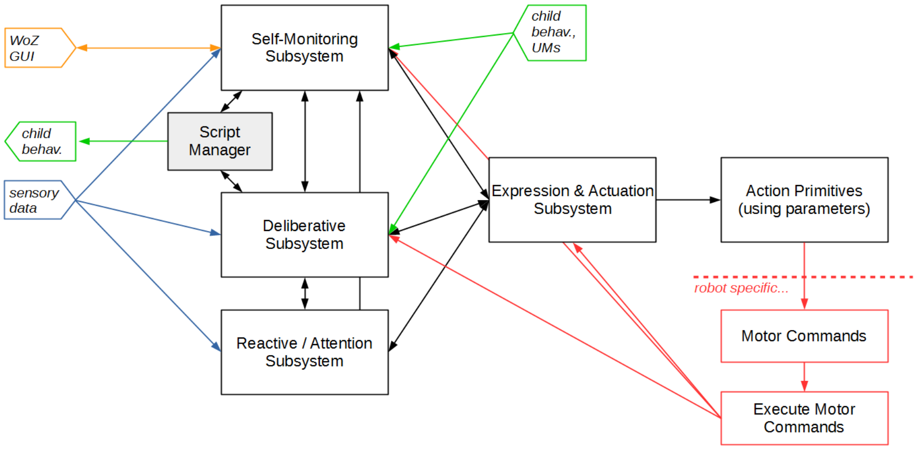


Figure 2: Description of the cognitive controller subsystems. The script manager is separate from, but tightly interacts with, the Deliberative subsystem to enable the robot control system to generate appropriate social/interaction behaviour even in the absence of an explicit interaction script. UMs: User Models.

## 2 The Expression and Actuation subsystem

According to the DREAM project DoW, the goal of the Expression and Actuation subsystem is to translate the actions of the social behaviour into readable social verbal and non-verbal cues, especially for our particular audience of young users with ASD. Since the specification is that all internal descriptions of behaviour are robot-neutral, this subsystem also has to be platform independent.

### 2.1 State of the art

A number of robots capable of gesturing have been developed to study different aspects of gesturing in HRI. Gestures implemented in robots are however, up to now, subject to two important limitations. Firstly, the gestures implemented in a robot are always limited to a set of gestures necessary for the current research, and often limited to one type of gestures. The robot WE-4RII [2] for example, was developed to study human-like emotion, hence, the incorporated gestures are mainly focused on emotional expressions. On the other hand, the developers of Robovie aimed for communication robots that interact naturally with humans. Since Robovie applications were focused on object indication and route direction-giving, mostly deictic gestures were used [3]. The reason for the limited amount of gestures implemented in specific robots can be found in the second limitation; namely the way gestures are implemented. Gestures are mostly preprogrammed off-line for the current robot configuration. The resulting postures are stored in a database and are replayed during interaction. This is the case for, among others, Robovie [4], HRP-2 [5] and Kobian [6]. Since the postures are dependent on the morphology, they are robot specific and cannot be used for other robots with other configurations.

Another common way to generate gestures is by mapping human motion capture data to the robot. This is for example the case for Repliee Q2 [7], where a marker-based motion capture system is used. Another possibility is to use the Kinect to perform skeleton tracking [8]. In [9], both a marker-based (Vicon) as a markerless motion capture system was used to reproduce human motion for the robot ARMAR-IIIb. Since the mapping of the captured data is robot specific, also these resulting gestures are dependent on the morphology and not usable for other robots. The result is that, when working with a new robot platform, new joint trajectories to reach the desired postures need to be implemented, which can be time consuming. It would however be much more efficient to make the implementation of gestures more flexible and to design a general method that allows easily implementing gestures in different robots, see figure 3.

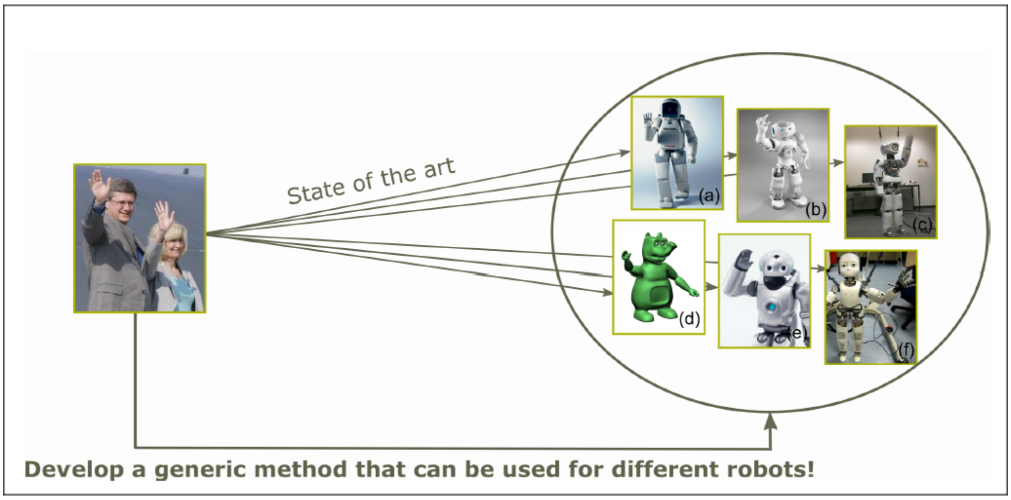


Figure 3: In the state of the art, gestures are always implemented for a specific robot platform. Our method aims to facilitate implementing gestures for a new robot platform by storing gestures independently of a morphology, and mapping them on a specific configuration. Robots: (a) ASIMO [10], (b) NAO [11], (c) Myon [12], (d) Probo [13], (e) QRIO [14], (f) iCub [15].

One of the approaches that flexibly generate gestures by different robots is based on neural networks, see [16]. However, this technique requires training. In both [10] and [17], a gesture framework initially developed for virtual agents is applied on a humanoid robot. In [10], the speech and gesture production model developed for the virtual agent Max is used to generate gestures for the ASIMO robot. Similarly to the ideas behind our subsystem, in [17], gestures are described independently of the embodiment by specifying features as the hand shape, wrist position and palm orientation. To generate gestures for the NAO robot, the correct angles for the shoulder and elbow joints are selected from a predetermined table listing all possible wrist positions and the corresponding joint values. The values for the remaining joints, namely the wrist joint and fingers are calculated by taking into consideration the values of other features such as the hand shape and palm orientation. So although the gestures are described independently of the robot configuration, mapping these gestures to the robot requires hard coded joint information. Our desired method should aim to fully automate the mapping of gestures to a random robot configuration.

Different robots use the Facial Action Coding System (FACS) by Ekman and Friesen [18] to abstract away from the physical implementation of the robot face. FACS decomposes different human facial expressions in the activation of a series of Action Units (UA), which are the contraction or relaxation of one or more muscles. In our work, we use FACS for the facial expressions and a similar framework for the rest of the body of the robot.

## 2.2 Research Objectives

The main functionality of this subsystem is to determine which combination of low-level actions the robot should execute next, and how these actions are to be performed. Suggestions for actions to take will come from three other subsystems: Deliberative, Reactive/Attention, and Self-Monitoring, see left side of figure 4. Along with this, it is assumed that the supervising therapist, through the GUI, will determine (either beforehand or in real time) the aspects of robot behaviour that should be executed, from which relative priorities will be determined for the three subsystems. This covers for example whether external disturbances (a loud noise in the background, or the appearance of a new face) should be reacted to by the robot (by leaving the script for a while for example), or ignored (with the script rigidly adhered to). The Expression and Actuation subsystem will combine these sources of information in an appropriate manner, see Motion Mixer in figure 4, ensuring that the stability of the robot is maintained. For example, if a greeting wave is requested by the Deliberative subsystem, and the Reactive/Attention subsystem wants to look at a face that has been detected, then the Expression and Actuation subsystem can combine the two by executing both (if the robot can remain stable by doing so). For a basic first step switches based on priority level could be used: i.e. if the script requests an action, execute it (and only it), but if there is no script action requested, then do what the Reactive/Attention subsystem proposes. However, the intention is to provide full behaviour mixing capabilities based on derived priorities from the therapists.

All this should be complemented by affective information, if this is available and appropriate to use. For example, the speed of motor execution could be related to arousal levels, or the choice of action sequence could be based on valence levels (if appropriate alternative sequences exist). This functionality will need to be switched on or off as required by the therapist based on child-specific considerations, and the relation to the therapy script (it may not appropriate to add emotional colouring to actions during the diagnosis procedure for example).

To approach such challenges, the first task should be to design a platform-independent representation of expressions. As explained above we have based our work on the Facial Action Coding System by Ekman and Friesen [18]. In a similar way, Body Action Units (BAU) will be defined together with a Body Action Coding System, where the different gestures are decomposed in the activation of BAUs. The BACS will point out the Action Units that need to be actuated for the generation of a desired gesture or body pose. This system avoids pre-programming of robot-dependent body poses and actions, which is relevant since humans are able to recognize actions and emotions from point light displays (so without body shape) [19].

The physical actuation of Action Units will depend on the morphology of the robot: a mapping will be needed between Action Units and physical actuators, this mapping will be specific to a robot platform and we will explore the possibility of learning this mapping. To translate this to the morphology of the robot, the Action Units need to be mapped to the degrees of freedom, and thus to the joints of the robot, see right side of figure 4.

A second task will be the categorisation of actions, comprised of temporal series of FACS and BACS, and the organisation in libraries that are accessible from the behaviour subsystems (Reactive, Attention and Deliberative). All actions for the different behaviours should be stored and expanded upon without the need to reprogram other subsystems.

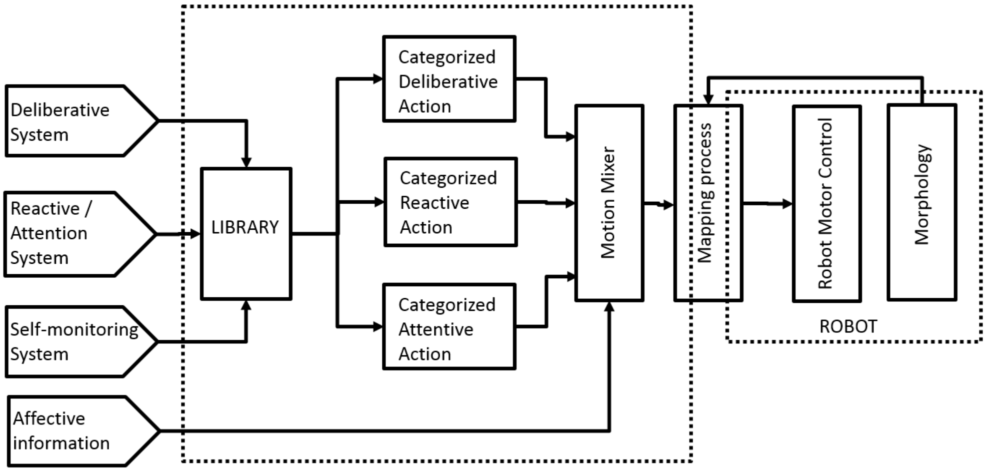


Figure 4: Overview of the Expression and Actuation subsystem. This subsystem receives inputs from several sources, categorizes them using the Library module and mixes them up to create a unique behaviour. Such behaviour is mapped into the joint configuration of the corresponding robot. This last process is done collaboratively between the subsystem and the robot.

### 2.3 Actual work performed

To generate expressions or gestures for a certain robot joint configuration, the developed method uses a set of target gestures listed in a database, which replaces the library of behaviours to be developed within the second task, and maps them to that specific configuration.

Our method divides the robot embodiment in three areas: the face expression, developed to provide the behaviours with natural and emotional features; the overall pose, developed to calculate gestures whereby the position of the main parts of the body is crucial; and the end effector, developed for pointing and manipulation purposes.

We [20] have already implemented the FACS methodology in Probo to express emotions, see figure 5. The Action Units (AU) are used to define the motions for Probo's DOF. As Probo does not have a human face and for simplifying the design, some of the AU were missing, others were replaced and some were added. To make the robot capable of expressing emotions, a two-dimensional emotion space based on the circumplex model of affect by Russell [21] was used. In the emotion space a Cartesian coordinate system was used, where the x-coordinate represents the valence and the y-coordinate the arousal, consequently each emotion $e(v, a)$ corresponds to a point in the valence-arousal plane. Each emotion can be represented as a vector with the origin of the coordinate system as initial point and the corresponding valence-arousal values as the terminal point. The direction $\alpha$ of each vector defines the specific emotion whereas the magnitude defines the intensity of the emotion. Each DOF that influences the facial expression was related to the angle of the emotion vector.

The NAO robot has not the facial expressibility that Probo has. It has no DOF in the face and the only mechanism that it has to express facial gestures is through the change of colors in its eyes. For such reason, an eyebrows system that will help to understand better emotional expressions on NAO's face has been developed, see figure 6 and Annex 3.3 for further details.

To ensure a realistic and readable overall posture, it is necessary to take into account the relative orientations of every joint complex the robot has in common with a human. A base human model

Figure 5: FACS has already been implemented in the huggable robot Probo to express different emotions.

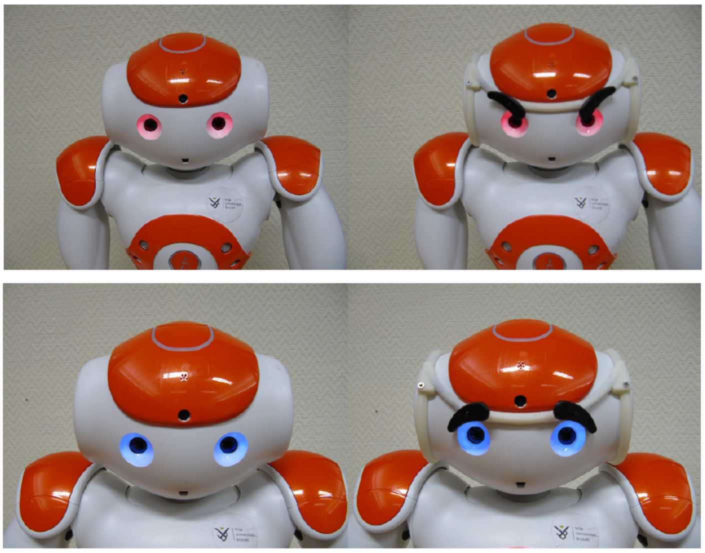

Figure 6: Comparison of the NAO robot expressing anger and sadness with (right) and without (left) the eyebrows system.

was defined, and the target postures were quantitatively described by the orientation of the different joint complexes in the model using the Body Action Coding System (BACS). As explained above, this is similar to the Facial Action Coding System of Ekman and Friesen [18], in this case a set of Body Action Units (BAU's) is defined. While the Facial AU's are defined as a muscle or a muscle group, our BAU's are based on the human terms of motion. The units are grouped into different blocks, corresponding to one human joint complex, such as the shoulder or the wrist. These blocks can subsequently be grouped into three body parts, namely the head, body and arm, which we refer to as chains. In that way, a base human model was defined, consisting of four chains; the head, the body, the left arm and the right arm. Although the leg movements also contribute to the overall performance of the gesture, for a first validation of the method we decided to focus only on the upper body movements.

As said before the Expression and Actuation subsystem outputs the gesture to-be-performed by the robot. Such gesture or behaviour is the result of the combination of several suggestions made by the other subsystems. Once we have the activated FAUs and BAUs, the subsystem should mix them using the Motion Mixer in figure 4. We have already made some developments regarding this module. Based on insights from the animation industry it combines motions commands that are triggered by a human operator with motions that originate from different units of the cognitive control architecture of the robot, see [22] for further details.

To make a certain model or robot perform a desired gesture or behaviour, this information should be mapped to its joint configuration. To specify the robot's joint configuration in the program, the Denavit-Hartenberg (DH) parameters of every present block need to be specified. A target posture is mapped to the configuration by imposing the orientation of the end effector of the different blocks and calculating the corresponding joint angles. This step has only been developed for the overall posture, which has been successfully validated on the virtual model of different robots through a survey, see figure 7. See Annex 3.1 for further information.

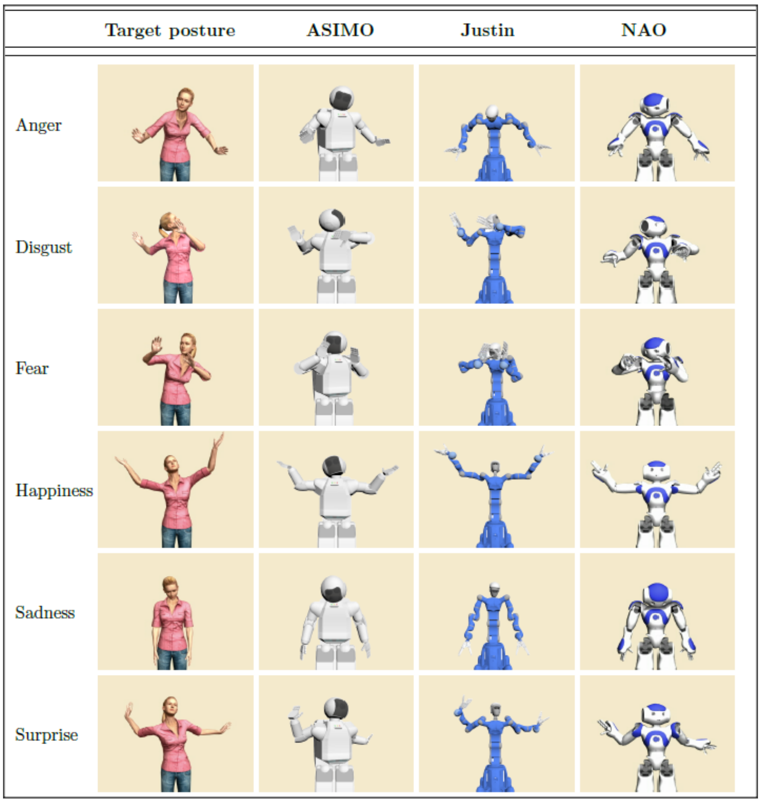


Figure 7: End postures of the gestures used in the survey. The first column shows the end posture of the target gestures for expressing the six basic emotions, while columns 2, 3 and 4 respectively show the mapped end postures for the robots ASIMO, Justin and NAO.

### 2.3.1 Action Primitives and Motor Execution

Once the desired gesture has been mixed and mapped into the corresponding joint configuration of the robot. The low-level actions are to be executed by the robotic platform.

The execution of these low-level actions is handled in a number of steps, as outlined in the "Robot Low-Level Motor Control" technical report, see Annex 3.4. This provides an interface between the control system (handled in a Yarp-based system) and the API of the robot hardware (Naoqi in the case of the Nao). The purpose is both to provide a bridge between the two systems, and to provide information to behaviour planning and supervisory oversight regarding the progress of motor command execution, including why a fail occurs if it does. This can be used to inform future action selection for example (by providing feedback for learning). The preliminary version of this component has been implemented and is running as part of the turn-taking WoZ system deployed as part of T2.1.

In addition to this low-level control system, there is the possibility that hardware abstraction can be handled automatically: i.e. that motor commands at the joint level can be determined automatically for different robot embodiments, without having to manually encode each specific action.

### 2.3.2 Preliminary Expression and Actuation Subsystem Component

Based on the functional description of the cognitive controller system of the DREAM architecture (see "*Organisation of Cognitive Control and Robot Behaviour*" annex), a preliminary implementation of the Expression and Actuation subsystem has been formulated. This first version of the component is defined in terms of the input and output ports, following the guidelines established in the software engineering standards (WP3). This is directly informed by the development of the WP6 control architecture in Y1, where each subsystem was defined in terms of the interactions with other subsystems, and their functions as outlined in the DREAM DoW. Please refer to figure 1, above, to provide this context.

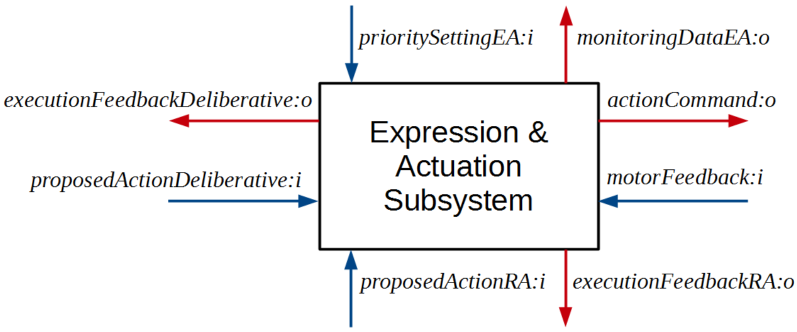


Figure 8: Expression and Actuation subsystem component yarp ports: the prefix for the port names is listed in the main text. The functional description of this component may be found above.

A description of the ports of this preliminary version of the component may be see in figure 8. As per the software engineering standards, each of the port names shown is prefixed by the system and subsystem names. For the script manager, this prefix is "*/cognitiveController/expressionActuationSubsystem/*". In this preliminary version, each of the ports shown are of type *BufferedPort<Bottle>*. This affords maximum flexibility in the type of information that can be sent to/from this component (and indeed the others), although this then requires that a protocol is defined for the exchange of information within the *Bottle*.

# References


[1] Serge Thill, Cristina A Pop, Tony Belpaeme, Tom Ziemke, and Bram Vanderborght. Robot-assisted therapy for autism spectrum disorders with (partially) autonomous control: Challenges and outlook. *Paladyn*, 3(4):209–217, 2012.

[2] Itoh K., Miwa H., Matsumoto M., Zecca M., Takanobu H., Roccella S., Carrozza M., Dario P., and Takanishi A. Various emotional expressions with emotion expression humanoid robot we-4rii. In *IEEE technical exhibition based conference on robotics and automation*, pages 35–36, 2004.

[3] Ishiguro H., Ono T., Imai M., Maeda T., Kanda T., and Nakatsu R. Robovie: an interactive humanoid robot. *Industrial robot: An international journal*, 28(6):498–504, 2001.

[4] Sugiyama O., Kanda T., Imai M., Ishiguro H., and Hagita N. Natural deictic communication with humanoid robots. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 1441–1448, 2007.

[5] Ido J., Matsumoto Y., Ogasawara T., and Nisimura R. Humanoid with interaction ability using vision and speech information. In *IEEE/RSJ International conference on Intelligent Robots and Systems*, pages 1316–1321, 2006.

[6] Zecca M., Mizoguchi Y., Endo K., Iida F., Kawabata Y., Endo N., Itoh K., and Takanishi A. Whole body emotion expressions for kobian humanoid robot: preliminary experiments with different emotional patterns. In *In the 18th IEEE international sympsium on robot and human interactive communication*, pages 381–386, 2009.

[7] Matsui D., Minato T., MacDorman K., and Ishiguro H. Generating natural motion in an android by mapping human motion. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 3301–3308, 2005.

[8] Tapus A., Peca A., Aly A., Pop C., Jisa L., Pintea S., Rusu A.S., and David D.O. Children with autism social engagement in interaction with nao, an imitative robot. a series of single case experiments. *Interaction studies*, 13(3):315–347, 2012.

[9] Do M., Azad P., Asfour T., and Dillmann R. Imitation of human motion on a humanoid robot using non-linear optimization. In *IEEE/RAS International Conference on Humanoid Robots*, pages 545–552, 2008.

[10] M. Salem, S. Kopp, I. Wachsmuth, and F. Joublin. Towards meaningful robot gesture. In *Human Centered Robot Systems*, pages 173–182. Springer, 2009.

[11] A. Beck, L. Cañamero, and K. A. Bard. Towards an affect space for robots to display emotional body. Ro-man, 2010.

[12] M. Hild, T. Siedel, C. Benckendorff, C. Thiele, and M. Spranger. Myon, a new humanoid. In *Language Grounding in Robots*, pages 25–44. Springer, 2012.

[13] K. Goris, J. Saldien, B. Vanderborght, and D. Lefeber. Mechanical design of the huggable robot probo. *International Journal of Humanoid Robotics*, 8(03):481–511, 2011.

[14] F. Tanaka, K. Noda, T. Sawada, and M. Fujita. Associated emotion and its expression in an entertainment robot qrio. In *Entertainment Computing-ICEC 2004*, pages 499–504. Springer, 2004.

[15] G. Saponaro and A. Bernardino. Generation of meaningful robot expressions with active learning. In *Proceedings of the 6th international conference on Human-robot interaction*, pages 243–244. ACM, 2011.

[16] C. Stanton, A. Bogdanovych, and E. Ratanasena. Teleoperation of a humanoid robot using full-body motion capture, example movements, and machine learning. In *Proc. Australasian Conference on Robotics and Automation*, 2012.

[17] Q. A. Le, S. Hanoune, and C. Pelachaud. Design and implementation of an expressive gesture model for a humanoid robot. In *Humanoid Robots (Humanoids), 2011 11th IEEE-RAS International Conference on*, pages 134–140. IEEE, 2011.

[18] Ekman P and Friesen W. *Facial Action Coding System*. Consulting Psychologists Press, 1978.

[19] A. P. Atkinson, W. H. Dittrich, A. J. Gemmell, A. W. Young, et al. Emotion perception from dynamic and static body expressions in point-light and full-light displays. *Perception-London*, 33(6):717–746, 2004.

[20] Jelle Saldien, Kristof Goris, Bram Vanderborght, Johan Vanderfaeillie, and Dirk Lefeber. Expressing emotions with the social robot probo. *International Journal of Social Robotics*, 2(4):377–389, 2010.

[21] Jonathan Posner, James A Russell, and Bradley S Peterson. The circumplex model of affect: An integrative approach to affective neuroscience, cognitive development, and psychopathology. *Development and psychopathology*, 17(03):715–734, 2005.

[22] Jelle Saldien, Bram Vanderborght, Kristof Goris, Michael Van Damme, and Dirk Lefeber. A motion system for social and animated robots. *INTERNATIONAL JOURNAL OF ADVANCED ROBOTIC SYSTEMS*, 11, 2014.

# 3 Annexes

## 3.1 Van de Perre et al (2014), Development of a generic method to generate upper-body emotional expressions for different social robots

**Bibliography** Van de Perre, G., Van Damme, M., Lefeber, D., and Vanderborght, B. (2014) Development of a generic method to generate upper-body emotional expressions for different social robots, *submitted to International Journal on Advanced Robotics for the special issue on humanoid robotics*.

**Abstract -** To investigate the effect of gestures in human-robot interaction, a number of social robots capable of gesturing have been designed. Gestures are often preprogrammed off-line or generated by mapping motion capture data to the robot. Since these gestures are dependent on the robot's joint configuration, they cannot be used for other robots. Therefore, when using a new robot platform with a different morphology, new joint trajectories to reach the desired postures need to be implemented. This method aims to minimize the workload when implementing gestures on a new robot platform and facilitate the sharing of gestures between different robots. The innovative aspect of this method is that it is constructed independently of any robot configuration, and therefore it can be used to generate gestures for different robot platforms. To calculate a gesture for a certain configuration, the developed method uses a set of target gestures listed in a database and maps them to that specific configuration. The database currently consists of a set of emotional expressions. The method was validated on the virtual model of different robots.

**Method -** To generate gestures for a certain configuration, the developed method uses a set of target gestures listed in a database and maps them to that specific configuration. To convey emotional expressions, the overall pose of the arms is important, and not (only) the exact position of the end effector. To ensure a good overall posture, it is necessary to take into account the relative orientations of every joint complex the robot has in common with a human. A base human model was defined, and the target postures were described by the orientation of the different joint complexes in the model. To make a certain model or robot perform a desired emotional expression, this information is mapped to its joint configuration. The necessary joint angles corresponding to the desired posture are calculated by using an inverse kinematics algorithm.

**Results and Conclusion-** The method was validated on different configurations, including those of the robots ASIMO, Justin and NAO, with arm configurations ranging from 9 DOF to only 5 DOF. The results are visualized by sending them to a virtual model. In general, the obtained postures are very recognizable, even for configurations with a limited number of joints. To validate the output of the method, an online survey was performed. The trajectories for the gestures corresponding to the six basic emotions were calculated for three robots, namely ASIMO, Justin and NAO and a separate movie for every robot performing each gesture and six additional movies showing a human virtual model performing the target gestures were made. The survey's objective was to investigate the quality of the mapped gestures, so to check whether the calculated gestures for different configurations are recognizable from the initial target gestures. The overall rates for the correct linking of the gestures are relatively high, whereof we can conclude that the calculated gestures in general well resemble the target gestures from the database and therefore, that our method to map gestures to different robot configurations gives good results.

**Relation to WP** This work directly contributes to Tasks T6.4.

### 3.2 Cao, H.L. at al. (2014), Enhancing My Keepon robot: A simple and low-cost solution for robot platform in Human-Robot Interaction studies

**Bibliography** Cao, H.L., Van de Perre, G., Simut, R., Pop, C., Peca, A., Lefeber, D., Vanderborght, B. (2014), Enhancing My Keepon robot: A simple and low-cost solution for robot platform in Human-Robot Interaction studies. *In* the 23rd IEEE International Symposium on Robot and Human Interactive Communication, Edinburgh, Scotland, UK.

**Abstract -** Many robots capable of performing social behaviours have recently been developed for Human-Robot Interaction (HRI) studies. These social robots are applied in various domains such as education, entertainment, medicine, and collaboration. Besides the undisputed advantages, a major difficulty in HRI studies with social robots is that the robot platforms are typically expensive and/or not open-source. It burdens researchers to broaden experiments to a larger scale or apply study results in practice. This paper describes a method to modify My Keepon, a toy version of Keepon robot, to be a programmable platform for HRI studies, especially for robot-assisted therapies. With an Arduino micro-controller board and an open-source Microsoft Visual C# software, users are able to fully control the sounds and motions of My Keepon, and configure the robot to the needs of their research. Peripherals can be added for advanced studies (e.g., mouse, keyboard, buttons, PlayStation2 console, Emotiv neuroheadset, Kinect). Our psychological experiment results show that My Keepon modification is a useful and low-cost platform for several HRI studies.

**Method -** Nonpolynomial Labs initiated the idea of hacking My Keepon with Arduino by reverse engineering. Based on this idea, we improved the firmware and developed a software to control My Keepon at the Vrije Universiteit Brussel (Belgium). Beatboxs afterwards offered the official source code to make the robotic toy completely hackable. My Keepon uses two microprocessors (PS232 and PS234) to control the movements and the sounds, which communicate with each other via $I^2C$ protocol. The PS232 (Slave - address 0x52) deals with sounds and encoders. The PS234 (Master - address 0x55) handles driving the H-bridges, detecting button presses, and main processing. My Keepon can be controlled by sending commands to these two microprocessors over the $I^2C$ bus.

**Results -** The modified My Keepon platform to some extent can achieve similar performances as of Keepon Pro in HRI studies. The modification method is simple and does not require advanced knowledge of electronics and programming. We organized a workshop of hacking Keepon at The 2013 International Summer School on Social HRI for a multidisciplinary group of students such as engineers, computer scientists, psychologists, etc. Even though many of them lack technical experience of soldering and programming, they were able to modify My Keepon in two hours. The workshop result proved that researchers can easily be familiar with the software, complementary Arduino shield and expand the platform. Hence, they can quickly set up the platform for their experiments.

**Conclusion -** We present a method to modify My Keepon to be a programmable research platform for HRI studies. Our website gives a complete tutorial with instructions for hacking the electronics and guidelines for software usage. Users are welcomed to modify the source code or integrate devices to fulfill their research needs or educational purposes. Our psychological experiments are used as examples of using the modified My Keepon in HRI studies. This work is expected to solve the current problem in HRI studies, i.e., the lack of low-cost robot platforms to enlarge the experiment scale or popularize the research results in society. Future work includes making the modified My

Keepon platform compatible with the Robot Operating System (ROS) software framework. With the advantages of ROS, developing software for robot will be easier thanks to ROS tools and libraries, as well as code sharing among researchers in the community.

**Relation to WP** This work directly contributes to Tasks T6.4.

### 3.3 De Beir, A. (2014), Eyebrows for Nao

**Bibliography** De Beir, A. (2014), Chapter 9: Eyebrows for Nao. *In Master Thesis:* An Autonomous Cognitive Architecture for Robot Therapy.

**Abstract -** In order to improve Nao's facial expressions. An eyebrow system has been designed and tested.

**Method -** An eyebrows system has been developed. Each eyebrow is actuated with a micro servomotor placed at the back of Nao's head. Both actuators are controlled with an Arduino Nano board which is connected to the Nao robot through an USB port. A Choregraphe box has been created to include the motion of the eyebrows as part of the programming of the robot. The whole system has been 3d printed.

**Results -** Tested by a questionnaire the eyebrows system improves the recognition of emotions in Nao's face. Just two emotions have been tested: anger and sadness. In the case of anger, it is been improved by 80.6 %, while for sadness it has been only by 32.7%.

**Relation to WP** This work directly contributes to Tasks T6.4.

### 3.4 Baxter, P. et al. (2014), Technical Report: Robot Low-Level Motor Control

**Bibliography**

**Abstract -** This technical report describes the first version of the low-level robot control system using YARP as the communications infrastructure. This system is designed to be extensible, and flexible to the requirements of the higher level robot behavioural components. A demonstrator system has been constructed for the Nao, but the structure is intended to be applicable to other robot embodiments (i.e. specifically the Probo, assuming a similar level of partially abstracted control is possible).

**Relation to WP** This work directly contributes to Task T6.4.

### 3.5 Baxter, P. et al. (2015), Technical Report: Organisation of Cognitive Control and Robot Behaviour

**Bibliography**

**Abstract -** The purpose of this technical report is to summarise the motivations and constraints underlying the cognitive control structures, and to outline an organisation of these subsystems. This is a proposal only; this document is intended to be a working one, to be updated as required during development. This version of the report is based primarily on the discussions that took place in Brussels (23/01/15).

**Relation to WP** This work directly contributes to Task T6.4, and the general organisation of the other systems within WP6.