Gualzru's path to the Advertisement World

F. Fernández, M. Martínez
Planning and Learning Group
University Carlos III Madrid
Email: {ffernand}@inf.uc3m.es

I. García-Varea, J. Martínez-Gómez
SIMD
University Castilla-La Mancha
Email: {ismael.garcia,jesus.martinez}@uclm.es

J.M. Pérez-Lorenzo, R. Viciana
M2P
University of Jaén
Email: {jmperez, rviciana}@ujaen.es

P. Bustos, L.J. Manso, L. Calderita,
M. Gutiérrez, P. Núñez
RoboLab
University of Extremadura
Email: {pbustos, lmanso, lvcalderita, marcog, pnuntru}@unex.es

A. Bandera, A. Romero-Garcés,
J.P. Bandera, R. Marfil
Ingeniería de Sistemas Integrados
University of Málaga
Email: {ajbandera, argarces, jpbandera, rebeca}@uma.es

***Abstract*—This paper describes the genesis of Gualzru, a robot commissioned by a large Spanish technological company to provide advertisement services in open public spaces. Gualzru has to stand by at an interactive panel observing the people passing by and, at some point, select a promising candidate and approach her to initiate a conversation. After a small verbal interaction, the robot is supposed to convince the passerby to walk back to the panel, leaving the rest of the selling task to an interactive software embedded in it. The whole design and building process took less than three years of team composed of five groups at different geographical locations. We describe here the lessons learned during this period of time, from different points of view including the hardware, software, architectural decisions and team collaboration issues.**

I. Introduction

Gualzru is a social robot built as an advertisement tool for a consortium of technological and digital media companies within the ADAPTA[1] project. The core of this project is an interactive panel able to provide personalized advertisement according to the preferences of the user. To achieve this goal, the consortium includes advertising companies, media asset management, software developers, technological consultants and software infrastructure providers, coordinated by the Software Labs group from the Spanish Indra company. The idea of using a robot as a more personal way of bringing people's attention was suggested in order to endow the panel with the ability to recognize the emotional state of the user and to classify her according to the estimated age and gender. After agreeing on creating a new social robot, it was decided that Gualzru, would team up with the interactive panel and boost the advertisement potential of the platform. The project started in May 2012 and this paper describes our experiences, successes and failures, during the three-year process.

II. The team

From the early analysis of the problem we knew that the project needed the expertise from different research groups. So, once a solid group of complementary researchers was agreed, we accepted to join the project's consortium. Our first 'robotic' consortium was composed by the Universities of Málaga, Extremadura and Carlos III of Madrid. From 2010, the first two groups were working together on the definition of a software framework for robotics, which could be used for the project. The University of Extremadura would also build the platform and would be the responsible of endowing the robot with the abilities for autonomous navigation and facial emotion detection. The University of Málaga would address the rest of vision-based problems (e.g. use facial descriptors to estimate the gender or age) and help with the navigation modules. Finally, the University of Carlos III of Madrid would be in charge of the high-level planning and learning modules. Everyone agreed in using this project as a test for the initial proposal from the University of Extremadura: to organize the whole software architecture around a centralized internal model of the outer world. Such representation is accessed by all software components to keep them informed about the current world state. They can also update it as the result of processing the data from the sensors. The ADAPTA project will provide a controlled but realistic scenario for testing the idea.

One major requirement of the proposal that the robot initially lacked was the ability to dialogue with people. To solve this problem the SIMD group from the University of Castilla-La Mancha, summing a large expertise on automatic speech recognition and natural language processing, joined the Consortium. Furthermore, the human-robot interaction ability was strengthen with the incorporation of the researchers from the University of Jaén. Six researchers from all groups were contracted during different periods of time to work on the project, however a larger group of researchers was always involved on the project.

A. Team coordination and sharing of resources

The coordination of this large group was supported by the use of collaborative tools. However, we soon understood that the only way to make a steady progress in the development of a large and complex project like ADAPTA, was by sharing a common code base and by scheduling periodic hackathons

[1] See Acknowledgments section at the end.

Fig. 1. Gualzru the robot

in which the members of all the teams could seat together for a week and fight a specific, common battle. This strategy naturally led to the division of the project in well defined milestones, consisting on system features to be integrated and tested during the one-week period. Coordination was therefore subtended on:

- A unique robotic prototype, available from month zero in a robotics simulator.
- A common programming framework, RoboComp [17], used in several previous projects. All the software developed for the project had to qualify as a RoboComp component, meeting the established quality standards, and had to be uploaded to a common git repository.
- A common cognitive architecture, RoboCog, available for all researchers and where individual modules could be inserted and tested minimizing the knowledge required about the rest of the architecture
- The organization of several intensive working weeks - hackathons- coinciding with the project milestones. These meetings were intense and dedicated to integrate and debug specific target functionalities.

To maintain the global view of the project and of the specific requirements, all the members should meet for each milestone and have always access to an open document storing this information. The document was edited online by all researchers and also served as a battlefield to discuss technical issues. We did not always coincide about how to do things but we agreed that the digital arena was the right place to fight.

III. Gualzru

Gualzru, a phonetic transcription of the English phrasal verb ”walk through” pronounced by a native speaker of Extremadura, is a 1.60m. tall robot with an external cover built of resin and fiber glass, and a differential base with two powered wheels and two casters. It includes gel lead batteries that provide an autonomy of three hours and all the necessary power electronics, recharging and power supplies for the sensors and processors. The complete fabrication of the robot was custom made by the groups of the consortium.

Table I shows the complex handcrafting process of Gualzru's external cover. This step was one of the most exasperating and time-consuming in the overall development of the project. It was a relatively new process for us with many steps that were out of our direct control. Going in Table I from top to botton and from left to right, we can rapidly summarize the manufacturing steps:

1) Gualzru's initial 3D design. To come up with a nice robot image we set up a public design contest among all Spanish universities and people and companies in the design business. One person from Cádiz, Spain, was selected among more than 30 proposals with a poll among a selected resolution committee.
2) The 3D drawings were sent to a company specialized in manufacturing expanded polystyrene molds using industrial CNCs. We learned that the choice of prices and qualities here are apparently important, since the final quality of the surface of the cover and the number of hours spent by the sculptor in fixing the small imperfections generated in the machining process were closely related. It is important to assure the final quality level in this early stage.
3) An external coating over the mold is necessary to facilitate the unmolding process. The mold is split in two halves.
4) A thick silicone layer is manually applied on the mold with additives to avoid sagging. This layer is called *negative*.
5) On top of the silicone a resin with fiber glass layer is applied to create a rigid external cover called *mother*.
6) Both layers are unmolded.
7) The silicone mold after being separated.
8) A *positive* mold is finally built by applying resin and fiber glass inside the *negative*. After drying, the cover is unmolded from the silicone and both parts are glued together. A final polish work is done to obtain a nice texture.
9) A solid and reliable differential base is built as the mechanical core of the robot.
10) The cover is fit on the base. Additional holes and slits have to be carved to allow for laser, camera, fastening, etc.
11) Sensors are incorporated to the robot. The tactile screen is placed after a final coating is ordered to a car painting workshop.
12) Gualzru at the University of Málaga in a public event with the University's Provost.

As a summary of the experience it is evident that the process is slow, expensive in working hours and almost impossible to rectify if a new idea comes by. The whole process took us many more months than expected and we had to use a replacement Nomad 200 robot while the robot was being built. In summary, it is a valid solution to the cover problem but with the arrival of 3D printing technology, all chances are that future robot covers will be divided in pieces small enough to be printed in a modern 3D printer, and then assembled together. There are also new small companies starting to offer these kind of services.

TABLE I

A SERIES OF SNAPSHOTS OF THE BUILDING PROCESS OF GUALZRU. SEE TEXT FOR DETAILS ON EACH STEP AND THE CONCLUSIONS OBTAINED AFTER IT WAS FINISHED.

IV. ROBOCOMP

As commented in section II-A, one of the few things that were already clear when the project started, was the need of a common code base.

A big part of the group had been already working in previous projects together and sometimes with other partners. From these works we learned that one of the main causes that prevented the formation of a cohesive, long lasting group with a common goal was the fact that each one was coding their own programs on different frameworks or without one at all. There are many robotics labs around, still unable to organize and create a coherent code base that grows from the accumulated work of dozens of researchers. After some tough negotiations involving the different frameworks that the groups were using or planning to use, we agreed to use RoboComp. We believe there are several reasons that, in the hindsight, justify this decision:

- We keep the control of the core and thus, we decide when to change and when to hold. It looks like a contradiction but when some complex open source software is very soon used by thousands of people, its evolution freezes or slows down almost immediately. The reason is that the core decisions made at the very beginning cannot be easily changed without generating compatibility problems and versions nightmares. As an example you can look at the widely expanded Microsoft's operating system (Windows) and the relatively slow addition of new features with each release (mainly nothing on the core changes). This does not mean that good software cannot be used by many people, but that complex software that deals with new, changing, not very well defined sort of things, takes its time to settle down.
- RoboComp's component model has been evolving since its beginning and has the necessary complexity for our needs. Not more.
- The current communications middleware, Ice by ZeroC [14], is extremely robust. No complaints and a big thank you to an excellent open source project.
- New middlewares could appear in the future with some game-braking features. In that case, if you control the framework you can define a reduced set of communication primitives, like the ones proposed by Schlegel in his PhD thesis [1] and a set of data types, and write some interface code that makes you framework middleware independent.
- A code generator is mandatory so the generic part of the components is always the same, compiles without errors and keeps the required quality levels. Code re-generation might be trickier but there are several techniques. RoboComp splits the working part of the component in two using inheritance. The inherited part is always generated and the part that inherits is generated only the first time. A lesson learned here is that it is easier if all tools and technology in the framework use the same development language and environment. Better if it is the one that most of the users are familiar with. Otherwise the no common specific tool becomes a bottle-neck that might delay and affect other parts of the framework.
  Our initial code generator was created with the Eclipse ecosystem using the existing tools it provides for DSL designs. This tool turned out not to be easily adjustable by developers (since they mostly develop C++ and Python) and a heavy environment that would not exactly match the team needs. Therefore, we ended up rewriting a lighter code generator in Python using *pyparse* and *COG* so

everybody could collaborate in the natural evolution of the tool. Now, RoboComp's code generator generates also Python components, that are becoming more a more popular due to their simplicity.

- We have developed all the tools we needed, although there are always tools that we would like to have but we have not had time to code them. RCIS deserves a special mention, RoboComp's simulator, that it has been there almost since the beginning of the framework. By the time RoboComp started the only existing open source simulator was Gazebo and it was in its early versions. If you are building a robotics framework and have already decided on the communications middleware, the chances are that you want a simulator that *speaks* the same language as the components of the framework. Only doing so, the simulator would behave like a component or several components with all the advantages that come with that. Therefore we wrote RCIS using Open Scene Graph [20] and an initial scene specification language that we named InnerModel. Later on, we discovered with great joy that having our own simulator would immediately provide us with an *emulator*. That is, a simulator that could be run inside the architecture computing in super real-time future courses of action and predictions. That is now part of our new architecture CORTEX, which is still in development.

V. RoboCog

Initially, we addressed the ADAPTA project from a very specific point-of-view. That is, giving the use case, we translated it to a finite state machine and assigned tasks to software components or groups of them that we call agents. The idea of using a finite state machine to manage the whole use case was soon unbearable. The number of states and transitions grew with every bit of reality added to scenario. Even modern hierarchical and concurrent formalizations of state-machines [8] and ready to go implementations such as the Qt StateMachine Framework [9] did not offer enough flexibility and maintainability to risk a project with many potential implications for our future.

We thus decided to take the hard way to a fully fledged symbolic planning system, in charge of the automatic generation of those huge state machines. The Planning and Learning Group at the University Carlos III of Madrid had a very long trajectory in these disciplines and was the perfect match to provide the needed technology. The use case was translated into a PDDL domain specification [18] and several planning algorithms were tested for that domain. A separated interface was clearly defined between high and low level domains. High-level being the domain of logic attributes and predicates, and low-level the domain of behavior agents that receive parametrized calls to act and provide metric values for relevant variables of the world state. The interface layer translates between high and low level, so both worlds are kept communicated. Of course, it is also the main cause of the so called, symbol grounding problem [11].

This now familiar scheme was synthesized by Erann Gat as the *three-layered architectures* [2], probably the most extended approach to build deliberative-reactive agent control systems today. However, when making decisions that directly involve human users, the domain of HRI, these architectures present some limitations. The most important one, from our point of view, is the need of a shared representation among all agents including metric and symbolic information, making each of them more aware of what was going on in the rest of the agents. For example, if a navigation module is driving the robot to a target place, and a person appears somewhere close to the planned path, how does the navigation agent differentiate between and obstacle and the person, so different avoiding (social) behaviors can be elicited? Or how does a conversational module knows that the person the robot is talking to, is not paying attention anymore, and thus a change in the discourse is advisable?

To us, it looks like that the good engineering practice of decoupling the problem in parts of *infinite impedance*, took away a crucial element, *context*. It is sometimes argued that context is somehow coded in the interactions between agents [3] or in the dynamics of coupled differential equations [4]. We decided to take here the more classical path of building an explicit shared representation for the context and face the problems to come.

To start, we already had a representation of the robot and its close environment in the form of a scene-graph, called InnerModel. This simple DSL served to initialize our RoboComp's 3D simulator, RCIS. Therefore, RoboComp's InnerModel was the perfect starting point to develop the idea of a shared representation of the robot, the environment and the people in it. The initial scene graph specification language was gradually extended to include more types of objects. Also a C++ class was written to hold in memory the graph and allow an easy and safe access to all the handy functionalities that this structure provided, such as coordinate transformations, measuring, insertion, modification and removing of nodes, perspective changing, frustum reachability, etc.

A basic scene-graph is essentially a kinematic tree with some add-ons. We had to incorporate all the symbolic information needed by the deliberative elements of the architecture. The requirements were that the perception-action related agents could update a fixed set of symbolic attributes and predicates, and that the selected representation could be efficiently translated to PDDL, so a specialized planning, executing and monitoring framework like PELEA [5] could be used. Our election on how to proceed in that situation was not exactly a mistake but it was certainly close to it. We decided to build a second graph, this time a graph rather than a tree, to hold this symbolic data and leave to a less stressed moment the problem of how to integrate both structures. It was not a mistake in the sense that the solution worked well and the robot managed to complete the use case. It was a mistake in the sense that now, months after the end of the project, we are hurrying to finish the integration of both structures because the separation is already generating many problems.

The current solution we are working on is the embedding of the kinematic tree inside the symbolic graph, and the code necessary to efficiently extract and insert the tree in a format that can be used by the many components that were written before the integration. It is hard to evaluate if the other choice would have permitted us to finish the robot on time, saving the posterior integration step. Software developing time is really hard to estimate, specially when robots are in the loop.

The new graph was named AGM, for Active Grammar-based Model [6], and besides fulfilling both requirements, it was also an experiment on planning with a variable number of symbols. In HRI, the perception component is getting more and more important. When interacting with a human in domestic or service environments, there are references to objects and places that may not be known beforehand. Of course, planning in an open world takes you out of the comfort zone of algorithms, where you know that the program will finish. In open worlds, there is always the possibility of adding a new symbol if the solution does not arrive.

AGM and InnerModel where independent structures and their coordination was managed by ad-hoc procedures, but AGM was finished on time and will maintain a complementary, symbolic representation of the robot and the objects in the scene. For example, a human in front of the robot being detected and represented as a skeleton inside InnerModel, could now be tagged *happy*, *focused* or *woman* in AGM. With AGM we had the missing part of the architecture and all the groups could start to meet in hackatons and reach, one by one, the urgent remaining milestones of the project. In summary, the main purpose of this dual representation was to provide both, a local description that could be updated and used by the different agents for their computations, and a shared context that is propagated among them to carry information that otherwise would remain hidden.

As a result of the graphs occupying their places with the agents, the overall idea of RoboCog started to change and we started to move from the three-tier original model to a non-hierarchical disposition in which all agents *gather around* these shared graphs and interact among them by reading, writing and propagating the changes. The abstraction axis is hidden inside the agents and defines what parts of the graph are accessible by its internal components. This is discussed in recent works by the group [7], [12]. Figure 2 shows a schema of the RoboCog architecture by the time of the final demonstrations of the project.

When a mission is assigned to the robot it is internally re-coded as a desired state in an AGM graph, which could include the whole world or just the symbols needed to satisfy the mission. The Executive module is the one in charge of achieving it. The steps needed to transform the current AGM graph are provided to the Executive by the *Task-based Planning and Monitoring* module PELEA as a sequence of tasks that are injected back in AGM. At this point, AGM holds the current belief about the world and the current desire about how the world should be. Agents scan the graph and find tasks that can be performed by them. Inside each agent there may be some limited capability of planning or sequencing sub tasks, e.g., maintain the interest of the person through dialog, monitor the correct execution of a gesture, recognize her facial emotions, etc. The most basic components are in charge of sensor motor loops and normally execute their commands without interaction with other components.

VI. The use case

We now describe the use case that constituted the main goal of the project. It was defined in the ADAPTA's kick-off meeting. The first version of this use case is depicted in Figure 3. It can be textually described as,

> Gualzru is waiting in the Waiting area. It is now ready to start one of its tedious working days. The Waiting area is at the middle of an uncluttered corridor in a large shopping center. People usually enter this side of the mall from the left side of the corridor, crossing in front of the Panel area before entering the shops. People going out the mall also cross in front of the Panel area, but walking toward the left part of the corridor. The objective of Gualzru is to offer products and services to all these people. In fact, its aim is to drive potential consumers to an advertising panel, in which these products and services will be displayed. As there are products for everybody, it can choose any person in the corridor. When it chooses a target, it moves from the Waiting area following an intersecting trajectory with the person's heading direction. This displacement is very short (2-3 meters maximum) and allows Gualzru to wait for the person in a static pose, facing her at comfortable social distance (1,5-2 meters minimum). Therefore, Gualzru can say 'hello' to the person without scaring her even if she is not very used to interact with a moving robot.
>
> If the person engages with him in this first contact, Gualzru will classify her into a group -using gender and age parameters- and will choose a product topic to offer. Product topics provide Gualzru an specific theme of conversation before inviting her to walk back to the Waiting area. During this short conversation, Gualzru will be always ready to say goodbye to the user if she shows the intention of leaving the conversation or if the presented product topic is not interesting to her. On the other hand, Gualzru must also check its batteries level to say goodbye and move to the Charging area if this level is under a minimum value. The Charging area is close to the Waiting area. In fact, when Gualzru arrives to the Waiting area, he will home itself to the Charging area. If the person agrees on going with Gualzru to the Waiting area, both move there and the robot says goodbye to her. Then, it returns to the Waiting area and waits for some time, which is the expected time that the person is going to be at the panel, before starting the process to select a

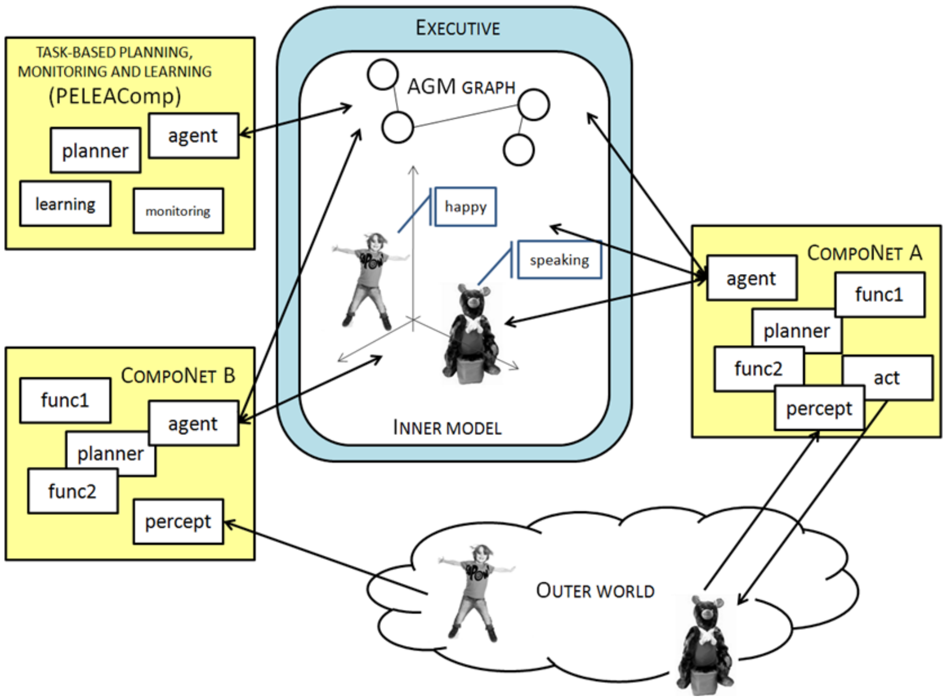

Fig. 2. An overview of the RoboCog architecture (from [16])

new target. As before, if batteries level are under a certain value, Gualzru moves to the Charging area for a reload.

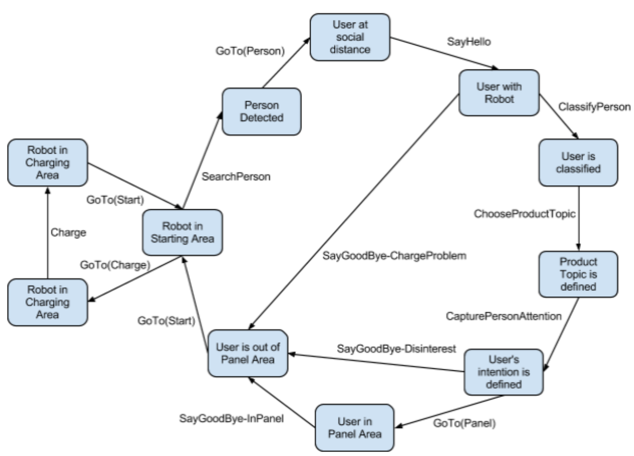

Fig. 3. The ADAPTA use case

VII. An Annotated Diary

As was mentioned before, the coordination of the project was based on periodic hackathons. We think this decision was a real success. We have already noticed the difficulty to integrate complex software and reach milestones without a real motivation from the people working in different labs. Many times, the global objective or the potential implications of the work are not correctly perceived. Other, personal relations get in the way. Hackatons have turned out to be an effective way to code, debug, test, share, make progress and build a team spirit.

A. May 2012. Kick-off

The project initiated with a kick-off meeting at Málaga where the overall strategy was discussed and the periodicity of the meetings was set.

B. December 2012. The "WORST" workshop at Cáceres

The main objective of the first hackathon was to explain and establish RoboComp as the common code base. All groups on the consortium had certain degree of knowledge about RoboComp, but it was considered mandatory to organize a workshop where simple examples could be programmed by all researchers under the supervision of experts from the Universities of Extremadura and Málaga. Fifteen people from all research groups and some more and some from Indra Software Labs met at Cáceres.

C. May 2013. First public demonstration at Málaga

For the first public demonstration of the whole project we had a simple prototype of Gualzru (Figure 4). Two autonomous behaviors were tested: the reactive navigation and a face detector. The AGM graph and the kinematic tree were able to internalize the perceived information. The seed of the architecture was planted. Obviously, not everything worked properly. The algorithms underneath both behaviors were changed in the final version, for example. But this fact was rather usual during the project. Other issues were more time consuming as expected. During 2013 we tried to replace the laser by

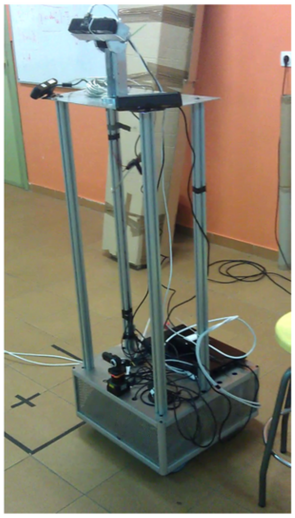

Fig. 4. The initial internal skeleton of Gualzru

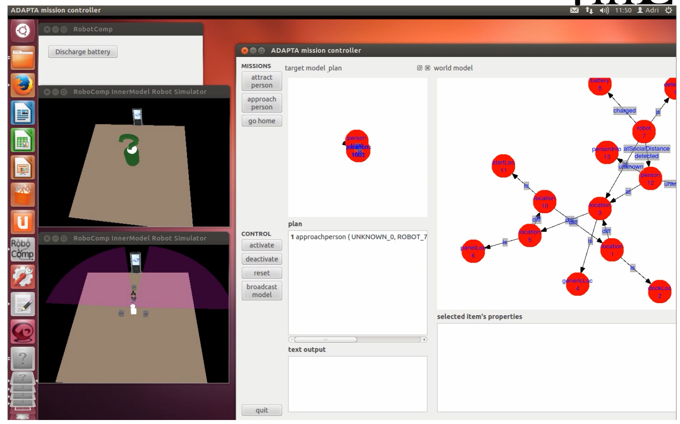

Fig. 5. Playing with the 'empty-boxes' architecture. The dashboard shows different panels: the one on the right shows the graph models that encodes (a) the goal to achieve -target model plan, and (b) the symbolic view of the outer world. It also includes the current action of the plan ('approachperson' in this figure). The panel on the left shows a visualization of the kinematic tree -up- and of the virtual world -down. We did not endowed the virtual robot with virtual sensors. The person is automatically detected if she is in front of the robot.

an array of RGBD sensors, arranged in a configuration that provided a wide field of view and theoretically gave good 3D coverage. Researchers from the Universities of Extremadura and Málaga were involved on achieving this goal. The sensor had limitations to perceive at short distances and we had to connect them to embedded computers like the Raspberry Pi of the time to liberate the USB ports in the main computers of the robot, Intel's NUC. We could not make the Asus' Xtion to run reliably with the available Raspbian. A few months after the end of the project, we succeeded with another board, the ODroid C1 [13], and now we can create hard-components that are cheap and provide real-time performance. We spent a few months trying to make that device work because we though it should work. Clearly, it was not the time. You have to choose the right battles.

D. November 2013. First evaluation of the architecture at Albacete

One of the major goals achieved after the meeting at Málaga was the development of a complete architecture able to work with a simulated robot in a virtual environment. The so-called 'empty boxes' architecture took this name from the fact that it included a complete version of the architecture RoboCog, although some of the components only had the public interface -IDL file- and the structure inside. Nevertheless, it included the two inner graphs -the symbolic and the geometric-, the conversational module, reactive navigation, person detector and high-level planning, executing and monitoring. To play with it we did not need the physical robot, but a computer with a RGBD sensor, speakers and a joystick. The simulated robot operated as an autonomous agent and we were able to move a virtual person in the simulator using the joystick. When robot and person were at interaction distance, the robot tried to convince the person to accompany it to the advertisement panel using his conversational skills. The speakers and the microphones on the RGBD sensor were used to support this interaction stage. The RGBD sensor was used to detect the face of a real person during the conversation. It was an intensive integration task.

One of the major successes of this architecture was the development of the triangle, high-level decision maker - executive - symbolic graph model. We were now able to translate to PDDL the information stored and updated in the symbolic graph (AGM) to the PELEA framework at the deliberative level. Furthermore, the Executive module was able to publish the graph to all software components on the architecture when a change was introduced. These components were arranged on networks, connected to the Executive through one distinguished component, the so-called agents. These agents were the responsible of maintaining the coherence of the information stored in the inner model, since they update the graph-model and, simultaneously, the geometric information of the kinematic tree. This second route was not supervised by the Executive. For the first time, the new definition of agent, included formally in the RoboComp component model and code generator, allowed all participants to share the graphs using the same interface. Our shared global representation on the state was now real and working.

We were able to launch more than fifteen software components. From this point of view, we were able to modify or add new components over a full-integrated architecture. Each successive meeting would imply a refinement of the previous proposal. While waiting for the robot Gualzru, see Section III for reasons explaining the long wait, we set up an old Nomad 200 robot with the RoboCog architecture and organized a new meeting at Albacete. See Figure 6.

At Albacete we evaluated for the first time the robot's behavior through questionnaires filled by the people interacting with the robot. The questionnaire is designed as a Likert scale, although it uses six levels, from 0 to 5, to remove the

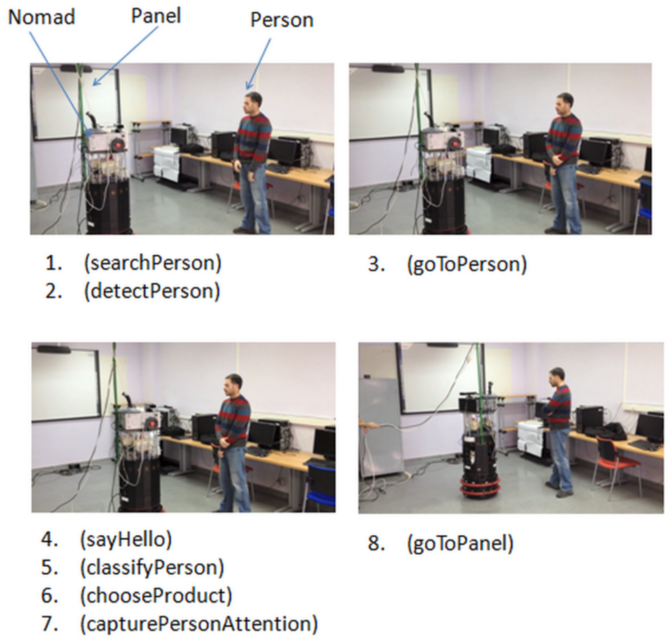

Fig. 6. The old Nomad performing through the use case at Albacete

neutral option -middle point. It is similar to that employed by Joosse et al. [15] to generate the database BEHAVE-II. Its main difference is that it has been created not from the point of view of the person observing the behavior of the user against the presence of the robot, but from the point of view of the same user that interacts with the robot. In this sense, we can consider that it collects influences of questionnaires of the Almere original model for man-machine interaction. In particular, the questionnaire includes a collection of questions arranged in four blocks: navigation, conversation, interaction and general sensations. The user fills the questionnaire giving a value for each response between 5 (completely agree) and 0 (completely disagree). These questions are listed in Table II. From that point on we would use that tool to evaluate the evolution of the project. Although at Albacete we were able to run the software in a real robot, we only could finish 12 use cases with different users. The number of questionnaires were also reduced to take hard decisions based on it, however the results were promising.

E. May 2014. Second public demonstration at Málaga

The Nomad 200 was moved from Albacete to Málaga and we continued testing specific problems related to the navigation module, speech generation and recognition, and person classification based on age and gender. The communication among components was forbidden and all information was transmitted through the inner representation. Things still did not work as we needed and the causes were not clear. Then, the robot Gualzru arrived and all efforts were translated to getting it ready.

For the second public demonstration Gualzru was already running RoboCog. Probably, this was not a mistake, but we did this integration without time to test the whole system. Also, we put the emphasis on collecting a larger collection of questionnaires. And to worsen things even more, the meeting was not set as the other previous hackathons: the goal did not focus on solving technical problems that were really there, but on showing a prototype that, at the end of the day, we should have known that it would not do the job. And the results were not good. During the demonstration the robot was able to interact with a person but it showed its limitations: the odometry alone was not able to correctly solve the localization problem, speech recognition had problems to work on crowed environments, the person classification module blocked the full use case until a good image of a face was taken, and so on. Another significant lesson was learned and never forgotten: you can not say that your robot will succeed in one trial until you have tested it for at least hundreds of times. There is a saying that can be applied here: ”let’s rehearsal so hard that the show looks like a rest”.

TABLE II
QUESTIONNAIRE RESULTS (50 TESTS)

Question	$\bar{x}$	σ
1.1 Do you feel safe when the robot approaches you?	4.31	0.95
1.2 Does the robot invade your personal space?	0.96	1.37
1.3 Do you think robot movements are natural?	2.62	1.23
1.4 Have you stepped away from the robot?	0.96	1.46
2.1 Have you understood the robot?	3.57	1.28
2.2 Has the robot understood you?	2.70	1.30
2.3 Was the conversation coherent?	2.96	1.38
2.4 Do you like the voice of the robot?	3.13	1.29
3.1 Did the robot get blocked?	1.39	1.72
3.2 Was the interaction natural?	3.11	1.11
3.3 Was the conversation fluent?	2.85	1.22
3.4 Did the robot seem to be tele-operated?	0.87	1.44
4.1 Did you enjoy the experiment?	4.31	0.88
4.2 Do you think the exp. was not interesting?	0.70	1.32
4.3 Would you like to repeat?	4.28	1.32
4.4 Would you recommend it to other people?	4.52	0.86

F. June 2014. Hackathon within the Workshop on Physical Agents

In June 2014, all groups had talks within the Workshop on Physical Agents (WAF2014) to be held in León (Spain). We asked the local organizers to facilitate us a working space to set up another hackathon during the week before. Vicente Matellán, the conference director gave us a cordial welcome and provided an excellent place for testing.

Before the hackathon, we discussed and organized the problems to solve there and when we arrived to León everybody knew what to do and joined in groups for a long week. The result was a real success: the dialog module was largely improved and tested, the localization problem was solved using AprilTags landmarks, and so on. The use case was repeated and repeated, and for the first time we detected real bugs and problems to deal with. After several days of intense work, Gualzru was able to do its job relentlessly until the battery was off! The robot spoke with all of us in the Lab and accompanied us to the panel. We had the impression that all ours problems were solved. We were happy for the moment...

But we were not going to enjoy the success for a long time. After discovering that the learning module would make the robot avoid people that always answered: 'No, I do not want to go with you to the panel', our host asked us to move Gualzru to the large hall where the conference was about to start and to have it welcoming the assistants. It looked like a good scenario for our use case. We accepted.

The moment we moved to the hall new problems appeared. The robot was unable to talk to people because nobody, not even humans, could hear what the other was saying. The space was wide open and we could not find a good place for the AprilTags landmarks. Light conditions were changing and the algorithms in charge of the RGBD camera did not always run correctly. For our younger researchers the experience was really hard, as they passed from the complete success to a glaring failure in a short time. Nevertheless, we were now in the final scenario. A spacious environment where the robot must interact with a specific person while other people are speaking and moving around. The failure had an aftertaste of an approaching victory. We still were able to close some use cases in this challenging scenario and a new time for improvements had started.

G. December 2014. Large evaluation test at Málaga

After a new demonstration at Ingenia (Málaga), in an environment very similar to the hall at León where we could capture new questionnaires, we returned to the Lab. The array of microphones of the Kinect sensor was intensively tested and, finally, we decided to change it for a shotgun microphone. As it is described with more detail in [12], other minor issues were also solved.

On December, 2014, the current version of Gualzru was tested in a real working scenario. The system was deployed in the hall of the Escuela de Ingenierías at the University of Málaga. The area where the robot was operating was about 70 square meters. Fixed obstacles included a column and some tables, but most of the area was free for the robot to move. The hall was populated by students and the trials lasted two half-days. The robot worked without human intervention and engaged with people passing nearby. These people had no *a priori* knowledge about the robot, nor its functionality. We collected a large set of questionnaires. The results are shown on Table II.

This data showed that the conversational system remained as the weak point of the robot. Some people did not correctly understood the robot due to the environmental noise and the voice of the robot was perceived as not particularly pleasant. But the most important issue was related to the understanding capabilities of Gualzru. Even when using the shotgun microphone these capabilities were strongly limited. The system is too sensitive to environmental noise and echos and it gets also confused when there are several people speaking around the robot. This situation is more common than expected due to the interest the robot produces. Additional issues such as different accents, voice volumes, etc. add more difficulties to the scenario. Despite these limited conversational skills, Gualzru achieved its main objective, to capture the attention of people. Most of them enjoyed the experiment and also would recommend the experience to friends or would like to repeat it. Comparing these results with the ones collected in the first experiments, revealed that successive updates in the robot have made it more robust and its conversational abilities, while still constrained, have been significantly improved.

H. Last stage: refining the HRI

The conversational abilities represented a severe drawback. Despite our efforts, only 50 % of the people that interacted with the robot in these real scenarios thought that it was able to maintain a coherent conversation. This was not enough for a robust, useful robot. But if you cannot solve a problem, perhaps is a good option to totally change the way to solve it. The speech recognition issue is hard to solve in noisy and crowded environments, where even humans find difficulties in understanding each other. Therefore, our idea was to look for alternative methods to allow people communicate with the robot. Speech recognition was reinforced with the incorporation of a tactile screen installed on the chest of the robot. The verbalized phrases were now displayed on this screen and it was possible for the person to answer the robot by touching it. This way, Gualzru retained its conversational abilities but the new interfaces increased its robustness and reliability. Following this modification, a new set of questionnaires were collected on the same scenario at the University of Málaga.

These questionnaires showed us that the mean values related to questions 2.1 and 2.2 (Table II) improved from 3.57 to 4.27 and from 2.7 to 3.72, respectively. Additional changes on the whole architecture allowed the robot to successfully close 93 % of the started use cases (on December 2014, this rate was 81 %). Furthermore, the unfinished use cases were always caused by the large amount of people in the place that would prevent Gualzru from reaching the panel.

VIII. Conclusions

In this paper we have presented the long process of creation of Gualzru, the salesman robot built for the ADAPTA research project. Looking at the starting requirements, we can firstly conclude that the final version of the robot conforms with the goals and expectations that we and the companies in the consortium initially had. But it is also true that even more rewarding than Ursus has been the whole process of collaboration and the knowledge distilled during these years. It is not that common that basic research is taken close to the *production line* while all the intermediate steps are registered and analyzed as a means to improve both, the forthcoming research and the methods and ways to generate reliable technology, given a limited amount of resources. For us, it has been a productive experience, both personally and professionally, and the capacity of the group to approach new technological challenges has increased notably. We have learned something useful in every step of the project, from the handcraft manufacturing materials and steps, to the way humans are starting to look at the (social) robots.

From the point of view of the technology that has been created and used in the project, we reaffirm the initial idea of the need for a common code base that brings together the work of all researchers. We still need some adjustments in the protocols and some refinements in the technology, and even more conviction by some doubters, but at the end of the day we might well be in the right track. The cost of maintaining and improving a framework like RoboComp is compensated by the flexibility of adapting it to your needs. Making good choices in this field, where Robotics meets Software Engineering, is not easy at all but once the software reaches a certain point of maturity, the leverage is undeniable. In the near future, we believe that these frameworks will play a crucial role in the evolution of intelligent robots. A role much more important that it is given today. It is needed the confluence of interested people from Software Engineering to gradually introduce new advances in DSLs, meta-models, model-based design and communication middlewares. From the recent evolution of social robot software, it looks like to us that the near future will bring larger and finer-grained networks of components, hundreds within the next years, that will demand more efficient software communications, self-diagnosis and repair, and sophisticated monitoring and deployment systems. Maybe classic, coarse-grained architectures will meet fine-grained ones at a point where interaction dynamics play a relevant role.

The cognitive robotics architecture, RoboCog, is a much more experimental and uncertain piece of handcraft. We started with a standard three-tier schema and managed to integrate symbolic planning with a fair amount of perception and action. To get there we re-introduced the idea of a shared representation among modules playing the role of an explicit context. It was implemented as two graphs, one geometric and one symbolic, and it proved enough for the required advertisement scenarios. Also, the introduction from the beginning of symbolic planning and learning technology in the project has proven a huge success. The initial idea of a using a flat PDDL description of the domain with a standard planner has evolved now into HRI specialized schemes, where hierarchical planners take care of quotidian, repetitive tasks and flat ones of the fine details and contingencies that might occur [19]. But, each solution takes to the next problem and before the end of the project, we were already working on integrating both graphs, reordering the classical hierarchies into more versatile organizations, infiltrate lifelong learning into all crevices of the system or use domain specific symbolic planning in classical low-level modules like navigation or object recognition. This issue, dealing with the overall organization of robotic intelligence, is undoubtedly the hardest one but projects like this motivate, and ultimately enforce, the search for new theoretical perspectives.

Other crucial part of the global Gualzru experience has been the use of evaluation metrics. User questionnaires turned out to be very important to improve the people's attitude towards the robot, as well as to reveal the most urgent weaknesses in preliminary stages. It is a valuable lesson to be kept that periodic tests and surveys are an important part of HRI research, although they are often seen by roboticits as a dull, questionable use of the scarce human resources available. Another important source of feedback are the robotic contests like RoboCup@Home[2] or RoCKin[3], that put all teams in the track of a common goal, and where real performances are evaluated in front of expert judges.

The ADAPTA project officially finished on May 2015 with a final public demonstration in Málaga. There Gualzru was able to interact with many people and successfully closed several difficult use cases. All partners in the consortium were satisfied and the robot will be serving from now on at the headquarters of Indra in Madrid. The research groups are now involved in more collaborative projects that hopefully will fund the construction of new social robots. We hope that the next generation will be capable of providing a better service to humans.

Acknowledgments

This paper has been partially supported by the Spanish Ministerio de Economía y Competitividad TIN2012-38079 and FEDER funds, and by the Innterconecta Programme 2011 project ITC-20111030 ADAPTA.

References

[1] C. Schlegel, "Communication patterns as key towards component-based robotics," *Advanced Robotic*, vol. 3, no. 1, pp. 049–054, 2006.
[2] E. Gat, "On three-layer architectures," in *Artificial Intelligence and Mobile Robots*, D. Kortenkamp, R. P. Bonnaso, and R. Murphy, Eds. AAAI Press, 1998.
[3] R. Brooks, "Intelligence without representation," *Artificial intelligence*, vol. 47, no. September 1987, 1991.
[4] C. ELIASMITH, "The third contender: a critical examination of the dynamicist theory of cognition," in *Mind Readings: Introductory Selections on Cognitive Science*, P. Thagard, Ed. MIT Press, 1999.
[5] C. Guzman, V. Alcazar, D. Prior, E. Onaidia, D. Borrajo, J. Fdez-Olivares, and E. Quintero, "Pelea: a domain-independent architecture for planning, execution and learning," *Proceedings of ICAPS'12 Scheduling and Planning Applications woRKshop (SPARK)*, pp. 38–45, 2012.
[6] L. J. Manso, "Perception as stochastic sampling on dynamic graph spaces," Ph.D. dissertation, 2013.
[7] R. Marfil, L. V. Calderita, J. P. Bandera, L. J. Manso, and A. Bandera, "Toward social cognition in robotics : Extracting and internalizing meaning from perception," *Waf2014*, no. June, pp. 1–12, 2014.
[8] *http://www.w3.org/TR/scxml/*
[9] http://www.qt.io/
[10] http://www.ros.org/
[11] Harnad, S. The Symbol Grounding Problem. Physica D 42: 335-346. 1990
[12] A. Romero, L. Calderita, J.P. Bandera, J. Martnez, R. Marfil, L.J. Manso, P. Bustos, A. Bandera. The Cognitive Architecture of a Robotic Salesman, in review.
[13] http://www.hardkernel.com/main/products/prdt_info.php
[14] http://www.zeroc.com
[15] M. Joosse and A. Sardar and M. Lohse and V. Evers, BEHAVE-II: The revised set of measures to assess users' attitudinal and behavioral responses to a social robot, International Journal of Social Robotics 5(3): 379-388. 2013

[2] http://www.robocup.org
[3] http://rockinrobotchallenge.eu/

[16] Calderita LV, Manso LJ, Bustos P, Surez-Mejas C, Fernndez F, Bandera A. THERAPIST: Towards an Autonomous Socially Interactive Robot for Motor and Neurorehabilitation Therapies for Children JMIR Rehabil Assist Technol 2014;1(1):e1
[17] Gutirrez, Marco A., A. Romero-Garcs, P. Bustos, and J. Martnez. "Recent Advances in RoboComp." Journal of Physical Agents 7.1 (2013): 38-47.
[18] McDermott, D. (2000). The 1998 AI planning systems competition. AI Magazine,21 (2).
[19] J. C. Pulido, J. C. Gonzlez, A. Gonzlez, J. Garca, F. Fernndez, A. Bandera, P. Bustos, and C. Surez, Goal-directed Generation of Exercise Sets for Upper-Limb Rehabilitation, in Proceedings of the 5th Workshop on Knowledge Engineering for Planning and Scheduling (KEPS), ICAPS conference, 2014, pp. 3845.
[20] http://www.openscenegraph.com