Integration of Humanoid Robots in a Collaborative Working Environment: A case study on motion generation

Olivier Stasse *AIST/CNRS JRL*
Rudolf Ruland *Fraunhofer FIT*
Florent Lamiraux *CNRS - LAAS* *AIST/CNRS JRL*
Abderrahmane Kheddar *CNRS - LIRMM* *AIST/CNRS JRL*
Kazuhito Yokoi *AIST/CNRS JRL*
Wolfgang Prinz *Fraunhofer FIT*

Abstract - This paper illustrates through a practical example a preliminary integration of a humanoid robotic architecture, with an open-platform collaborative working environment called BSCW. BSCW is primarily designed to advocate a futuristic shared workspace system for humans. We exemplify how such complex robotic systems (such as humanoids)can be integrated as a proactive collaborative agent who provides services and interact with other agents sharing the same collaborative environment workspace. Indeed, the robot is seen as a 'user' of the BSCW which is able to handle simple tasks and reports on their achievement status. We emphasis on the importance of using standard software such as CORBA in order to easily build interfaces between several interacting complex software layers, namely from real-time constraints up to basic Internet data exchange.

Keywords - Collaborative Working Environments, Humanoid Robots, OMG standards.

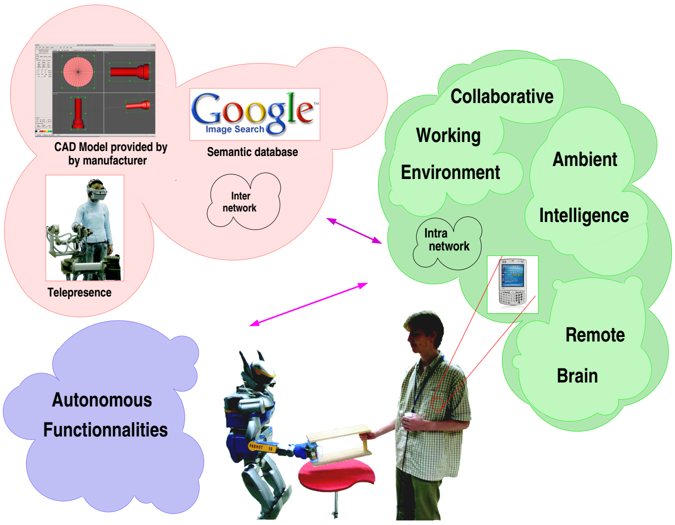


Fig. 1. Four contexts of task realization in the physical common workspace.

1. INTRODUCTION

Humanoid robots are currently targeted in several applications ranging from the house maid robot able to clean [1] or even cook [2], to industry fields as a multi-purpose robotic system which is flexible to fast changing in tasks and product lines, able to manipulate various products, inspect and guard small and middle size companies outside the factories, etc. In these scenarios it is important to integrate, and even to take advantages of the existing IT- infrastructure to each robot programming and mission assignments. The context of this work is a joint investigation between the European project ROBOT@CWE[1], which aims at introducing robots – especially humanoids– as agents in collaborative working environment, and another European Integrated Project eCoSpace[2] which purpose is to design a futuristic human-centric shared workspace system for advanced collaboration between humans. In a previous paper [3] we have described the current state of the art and limitations related to the integration of our particular humanoid robotic architecture in such environments.

The view taken in this paper is from the side of a software architect designer. In this context we have defined four contexts of task realization in a physical workspace as depicted in Fig. 1:

1) An *autonomous* context realization when the robot is directly interacting with a human to perform a task, and particularly during physical interaction.
2) A *local* context realization when the robot is using the surrounding network and computer capabilities to expand its functional space. This is typically the case in the presence of ambient intelligence and/or in the context of the remote brain approach [4].
3) A *semi-local* context realization when the robot is interacting with a collaborative working application targeted for an application or for a structure such as a company. It is semi-local because its semantic scope is local, but can be geographically spread over several locations.
4) A *global* context realization when the robot is interacting with services external to its *semi-local* structure for instance Google Images services, manufacturer product specification, etc.

This paper treats more particularly the semi-local context. We have demonstrated recently how two humans, the one in Japan collocated with the robot, and the other one being in Germany, could perform a collaborative tasks using a

---

[1] `www.robot-at-cwe.eu`
[2] `www.ip-ecospace.org`

telepresence system and a humanoid robot [5]. The person in Germany used a telepresence system to teleoperate a humanoid HRP-2 to lift an object with the operator in Japan sharing the same physical space and object with the robot. Although the realization of this experiment requires the use of complex control architecture in order to guarantee stability of the humanoid robot, and that of the overall system, the role of the robot was however limited to reproduce the actions of the master operator in Germany.

In a different context, Sagakuchi *et al.* [6] demonstrated how HRP-2 could be used in an intelligent house to perform autonomous actions such as closing the door. However if one aims at having humanoid robots used in working offices or flexible SMEs to perform various tasks and adapt quickly to fast changing products lines, the most efficient way to assign robotic tasks missions is to interface the robotic architecture to the specific working one if available. We demonstrate in this paper how HRP-2 relying on advanced architecture and planning software can be smoothly integrated into a real collaborative working environment. We also report on the use of software technology standards to easily build appropriate interfaces.

2. THE OVERALL ARCHITECTURE

The challenging part of this demonstration is to maintain separate the robotic technology from that of the overall collaborative technology which allows keeping their specificities. This is achieved by raising the functional level of the robot to a sufficient autonomy in a way it interacts with a human centric CWE architecture. The atomic understanding level robot and human 'languages' share is the *task* as commonly understood in the context of collaborative working environment and not in the control sense as introduced in section 3-2.2. Those tasks are defined in the context of BSCW, a collaborative working environment used by several European projects.

The semantic and the ontology of the task are not specified in this context. The task has merely a name, and several associated fields. The interpretation of those properties is left to the users to which those tasks are aimed for. Following this line, we give a brief overview of how the robot is able to interpret the task and reply with the appropriate answer.

To give a flavor of what the robot is able to do, we give an integrated overview of the experiments we have been able to achieve so far in the context of the project. Later on, a case study is described on a surveillance task for HRP-2 in a known environment together with experiments.

2.1. HRP-2's architecture

The Fig. 2 illustrates a functional block oriented architecture where each block is implemented by a CORBA server. The Low Level Vision server aims at providing early vision processing, in this example use it is limited to provide raw images at will.

The path planner is the component used to plan footprints and whole-body motion. The motion generator generates and realizes a dynamically stable motion when the robot needs to perform steps, or perform some tasks with its end-effectors. The decision layer is based upon the classical Hierarchical Finite State Machine paradigm. More precisely we are using the statechart specification of UML. The current extension of the standard template library called boost implement such specification. We have used it to realize the Decisional block. This part can be easily specified by a user using today's UML statechart modeler. In this paper, we will mostly describe the step planner server and the decision layer.

2.2. BSCW

BSCW is a cooperation platform on the Internet which allows sharing documents, organizing team's work by assigning tasks, organize meetings, create communities, allow direct communication or information distribution such as e-mails or RSS feeds. This creation of the Fraunhofer Institute for Applied Information exists since the mid-1990s, and is currently supported by a spin-off company called OrbiTeam. BSCW is being extended in the frame of the European Integrated Project called eCoSpace [7] to develop a collaborative environment for eProfessionals. The present paper is the result of collaboration between the ROBOT@CWE and the eCoSpace FP6 EC projects.

3. MOTION GENERATION, PLANNING AND HIGH LEVEL DESCRIPTION

3.1. Introduction

One of the main difficulties with complex redundant robot such as humanoids is to find a way to generate motions with simple enough instructions to be manageable by non expert human operators while maintaining the overall constraints inherent to the robot such as keeping balance, collision avoidance, reactive behavior, etc. In Neo *et al.* [8], they recently proposed such a system for on-line motion generator in the context of teleoperation. Mansard *et al.* [9] demonstrate how it is possible to generate motion autonomously using visual information. More recently Dariush *et al.* [10] used a similar approach to play back motion recorded on humans and adapt it to ASIMO. In the first case, the field of application deals mostly with cases where human assistance is needed e.g. to help people with limited mobility, in disaster situation, or in space application. In the second case, the approach is mostly reactive and need to be coupled with a higher decisional layer such as a motion planner. Yet, motion planners are still suffering computational complexity in finding trajectories, in the configuration space, to solve a routing problem. A key issue to find trajectories which are dynamically stable is to seek reduction properties, generally by simplifying models for the motion planner which correspond to the control architecture.

3.2. Motion generator

A. Stability: The stability criterion used in our work is the Zero Momentum Point which assumes that both feet are on a flat floor. This criterion is important because it reduces the set of trajectories that can be possibly be performed by the robot. Indeed when considering other stability criterion, the

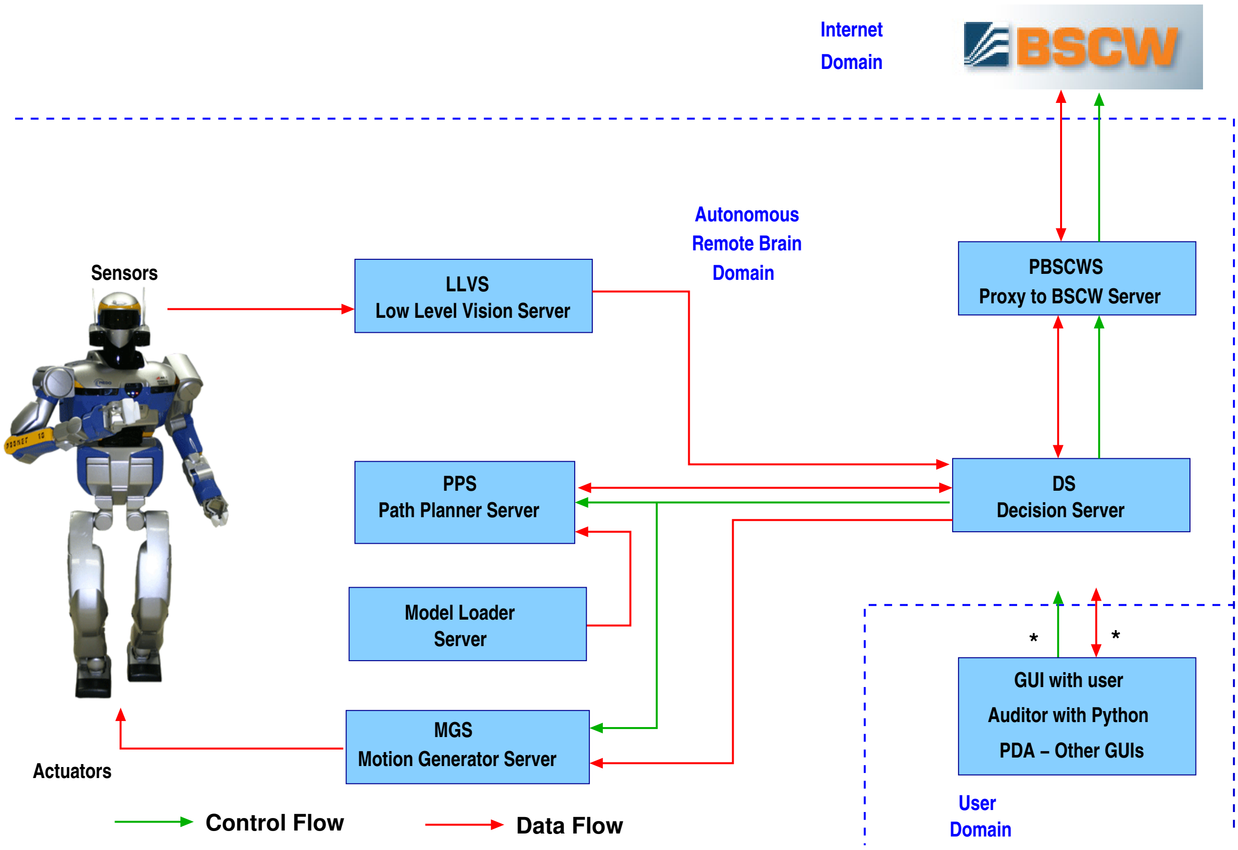


Fig. 2. Functional block implemented as CORBA and OpenRTM servers.

range of possible motions might include contact with obstacles and other complex interactions [11]. Moreover to make the problem tractable in the high speed control loop necessary for such robots, supplementary constraints are considered which simplify the numerical resolution, but also constrain the set of trajectories. The current general scheme used in humanoid robots acts more as dynamically stable reference generator and uses a simpler controller to track the reference. The algorithm implementations used to generate those references have been gathered in a framework allowing prototyping and multiple modalities [12]. This issue is specific to the robotic architecture.

B. Generalized Inverse Kinematics: Introduced initially by Nakamura, the generalized inverse kinematics (GIK) offers a prioritization scheme to associate several controllers together in order to generate motion for a redundant robot. Its equivalent in the force domain is the prioritized operational space control. These control concepts have been renewed in the context of whole-body motion generation in the field of humanoid robotics. Due to current practical limitations, most of the walking humanoid robots are not using a low-level torque control but rather a position-based control. Finding the activation and the prioritization of those controllers is still an open issue. Some work exists to use the GIK-approach in planning to correct trajectories when considering dynamical stability. For the sake of simplicity in the remaining of this paper, it is assumed that the underlying GIK provides one solution for one stable reference trajectory provided by the previous module.

3.3. Planning

Following the previous remark, current fast planning relies on simplified model which are known to be realized by the control architecture presented in section 3-2.2. A popular solution is to sample the set of feasible foot-steps of the robot, and perform an A^* search in the environment, see examples in [13][14]. In this work we propose a different approach where the robot is seen as a vertical bounding box moving in a plane. This model is used to connect configurations chosen by a probalistic roadmap configuration shooter. The software used in this paper is called Humanoid Path Planner (HPP) which is developed under the supervision of F. Lamiraux (the third author). It is build upon KineoWorks, a product commercialized by a spin-off company called KINEO[3]. This software provides the overall architecture to apply probabilistic roadmap with various robot models. In this specific application, the robot has three degree of freedom the position on the ground and the orientation. Once the configurations are connected with the chosen smooth function the steps are placed along the

[3] `www.kineocam.com`

trajectory. To generate the complete robot configuration, HPP uses the same walking pattern generator used to generate the motion of HRP-2 in the control architecture. This allows checking for undesirable collisions with obstacles. When no trajectory is found, and when the environment limits are known, the system is able to return a failure message.

3.4. Decision Layer

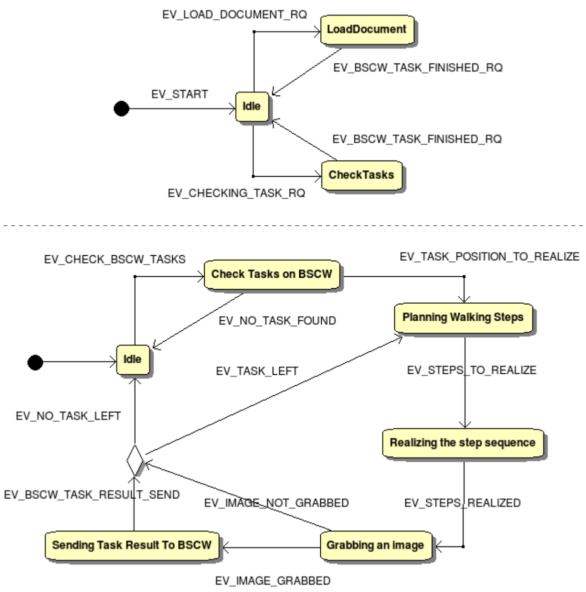


Fig. 3. Statechart model of the case study.

As it is done classically we used a Hierarchical Finite State Machine to map a discrete semantic with a set of controllers and parameters. This mapping is usually done in an arbitrary manner. Recent works [15][16][17] is trying to create automatically this mapping by grouping set of trajectories of human activities. There is an important issue here in making accessible the interfaces provided by the block depicted in Fig. 2 with a collaborative environment. In our case, this is filtered out by the decision layer.

One way could be to use the interface description of the component and expose them through Web Service Description Language. Fortunately the link described previously between the planning and the control layer allows the humanoid robot's high level system decide by itself if a motion asked to the robot is feasible or not. Such capability facilitates the user programming of the robot behavior, and frees the Collaborative Working Environment to have any knowledge on the robot.

4. SIMULATIONS AND EXPERIMENTS

Now we present our current achievement in integrating HRP-2 in a full-size CWE, such as BSCW.

4.1. Setup description

In order to achieve our integration of HRP-2 in a CWE, the hierarchical finite state machine depicted in Fig. 3 has been implemented to provide a simple decision layer. At first

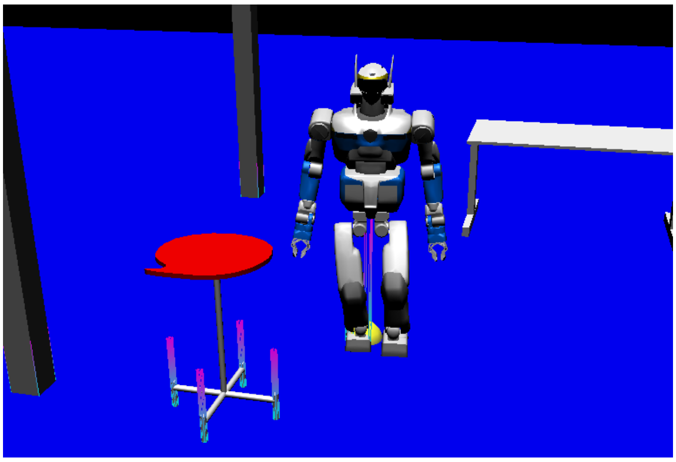


Fig. 4. Dynamical simulation of the steps generated by HPP.

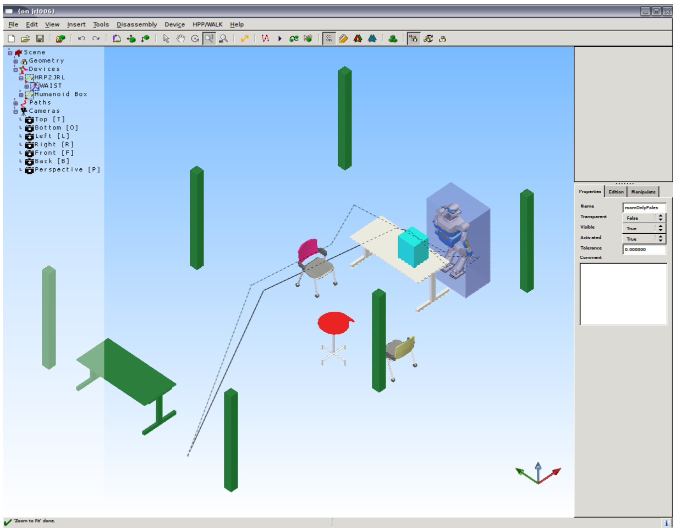
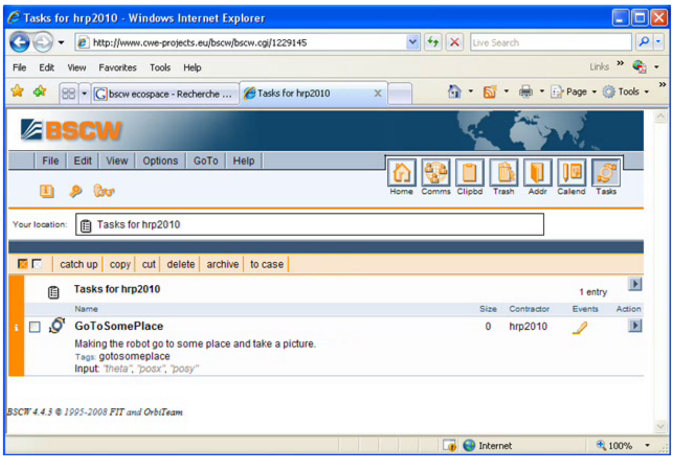


Fig. 5. HPP solving a more complex situation.



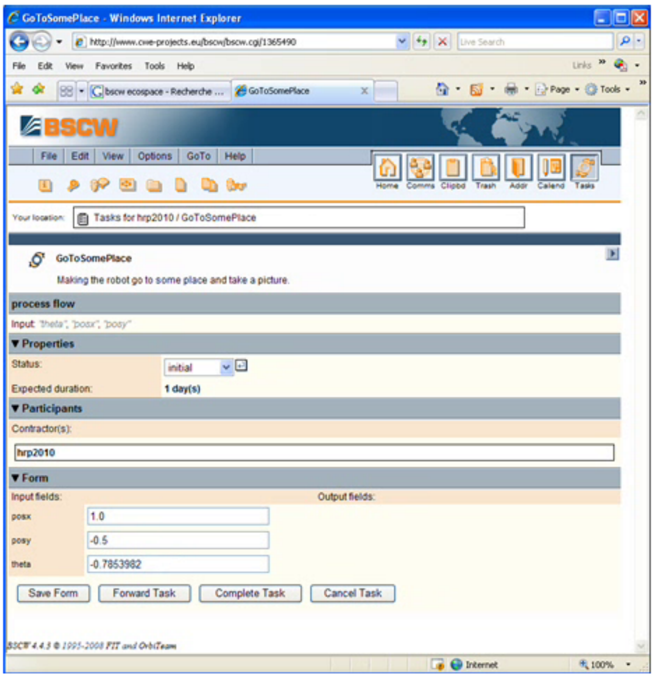
Your location: Tasks for hrp2010

Tasks for hrp2010 1 entry

Name	Size	Contractor	Events	Action
GoToSomePlace Making the robot go to some place and take a picture. Tags: gotosomeplace Input: "theta", "posx", "posy"	0	hrp2010		

BSCW 4.4.3 © 1995-2008 FIT and OrbiTeam

Fig. 6. List of tasks assigned to HRP-2 by others BSCW users.


GoToSomePlace - Windows Internet Explorer

http://www.cwe-projects.eu/bscw/bscw.cgi/1365490

Your location: Tasks for hrp2010 / GoToSomePlace

GoToSomePlace

Making the robot go to some place and take a picture.

process flow

Input: "theta", "posx", "posy"

Properties

Status: initial

Expected duration: **1 day(s)**

Participants

Contractor(s): hrp2010

Form

Input fields:

posx 1.0

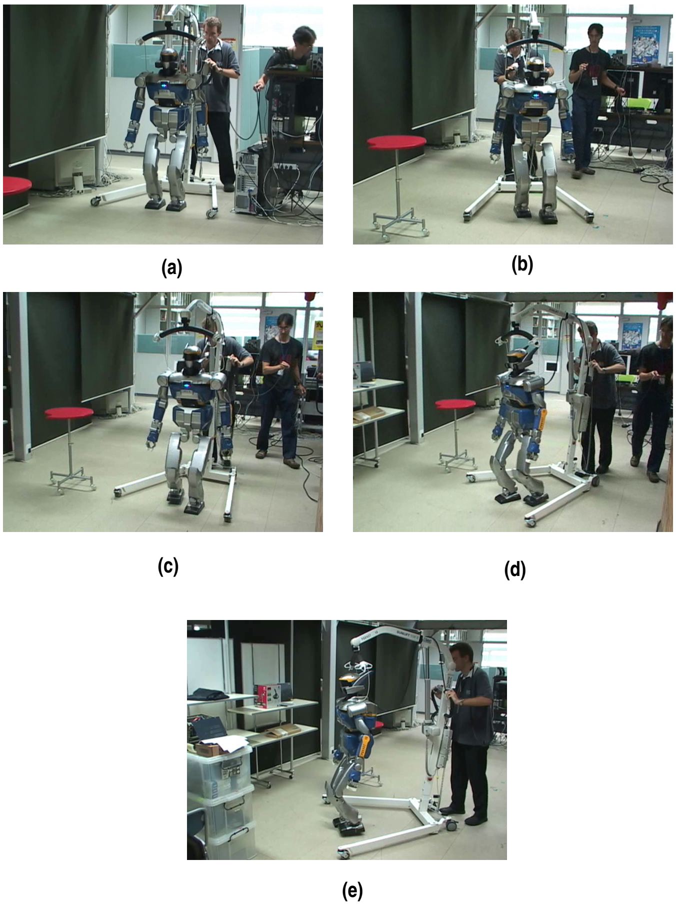
posy -0.5

theta -0.7853982

Output fields:

BSCW 4.4.3 © 1995-2008 FIT and OrbiTeam

Fig. 7. The task *GoToSomePlace* specified in BSCW.


(a) (b) (c) (d) (e)

the robot system connects to BSCW and identifies itself. It then checks in its list of tasks, Fig. 6 illustrates visually this process for a human agent, if there is any dedicated task. In our example, founds two tasks that are named *GoToSomePlace* and *TakePicture*, see Fig. 7, which has been assigned by another user of the shared workspace system. It then extracts from this task the attributes, i.e. the fields specifying the target position and orientation of it (the robot) to reach. From this target position and its actual position in the environment, assuming that the environment is fully known and static, the HPP software tries to plan a feasible trajectory for the humanoid. If such a trajectory exists, subsequent computed steps are sent to the robotic low level control architecture to realize the motion. Once the steps are realized the robot takes a picture and uploads it back to BSCW. Fig 4 shows a dynamical simulation of the steps generated by HPP in the case of non-cluttered environment. Fig. 5 displays a more complex situation handled by HPP. The simple situation has been executed on the real platform, and some snapshots of the experiment are depicted in Figs. 4-1.

4.2. Software consideration

To maximize the compatibility and the reuse of the software components, we make extensive use of standards, software tools and design patterns instead to concentrate on new concepts as much as possible. The control system and the physical simulation are realized using OpenHRP simulation and control software [18] which is currently supported by the Japanese government to become a national platform. Because of HRP-2 [19] embeds advanced CPU systems we are using mostly CORBA to handle the middleware issues. Because CORBA does not integrate any way to specify data flow, scheduling properties, control and interface parts of a component, a new OMG standard has been proposed called Robot Technology Middleware to fill the void. HPP, our planning framework, has been used together with this technology in this paper. CORBA and RTM made possible to use four machines with several cores to make the computation in a seaming-less manner.

With data flow structure, RTM [20] allows to avoid a dependency on interfaces and a graph can be constructed by an external client. When computational time is constant, it is possible with appropriately specified scheduling properties to perform model checking. The decision layer follows the UML statechart rationale and is implemented using the *boost::statechart* library [21]. We hope to move forward with an automatic code generation from model description.

The connection with BSCW is realized with XML-RPC, which allows using libraries already available to access the remote application. The open definition of a task in BSCW allows the robot to decide autonomously if the task is understood and feasible.

5. CONCLUSION

We presented our current achievement in integrated a humanoid robot in a real collaborative environment architecture using standard software and robotic technologies. With a sufficient level of functionalities, the robot is able to act as an autonomous agent (i.e. a CWE user) interpreting simple commands and sending a feedback to its *collaborators* (other agents, humans of machines) that are sharing the same workspace system. There are yet open issues with the mapping of the capabilities of such a robot in a company that make extensive use such advanced collaborative tools. We believe that raising the range of functionalities of such robot while using software standards certainly the right direction toward efficient robot integration in concrete applications.

6. ACKNOWLEDGEMENTS

This work is partially supported by grants from the ROBOT@CWE EU CEC STREP project `www.robot-at-cwe.eu` and the eCoSpace EU CEC IP project `www.ip-ecospace.org`, both under the 6th Research program.

REFERENCES

[1] K. Okada, M. Kojima, Y. Sagawa, T. Ichino, K. Sato, and M. Inaba, "Vision based behavior verification system of humanoid robot for daily environment tasks," in *IEEE/RAS International Conference on Humanoid Robots*, 2006, pp. 7–12.

[2] F. Gravot, A. Haneda, K. Okada, and M. Inaba, "Cooking for humanoid robot, a task that needs symbolic and geometric reasonings," in *IEEE/RAS Int. Conf. on Robotics and Automation*, 2006, pp. 462–467.

[3] O. Stasse, E. S. Neo, F. Lamiraux, A. Kheddar, and K. Yokoi, "Architectures and models for humanoid robots in collaborative working environments," in *International Symposium on Robotics*, 2008, p. accepted.

[4] M. Inaba, T. Ninomiya, Y. Hoshino, K. Nagasaka, S. Kagami, and H. Inoue, "A remote-brained full-body humanoid with multisensor imaging systemof binocular viewer, ears, wrist force and tactile sensor suit," in *IEEE/ICRA*, vol. 3, pp. 2497–2502.

[5] A. Peer, S. Hirche, C. Weber, I. Krause, M. Buss, S. Miossec, P. Evrard, O. Stasse, E. S. Neo, A. Kheddar, and K. Yokoi, "Intercontinental multimodal tele-cooperation using a humanoid robot," in *IEEE/RSJ International Conference on Intelligent RObots and Systems*, 2008, pp. 405–411.

[6] T. Sakaguchi, T. Ujiie, S. Tsunoo, K. Oohara, E. Neo, and K. Yokoi, "Intelligent-ambience robot cooperation - closing door with humanoid robot-," in *4th International Conference on Ubiquituous Robots and Ambient Intelligence (URAI)*, 2007, pp. 383–388.

[7] W. Prinz, H. Loh, M. Pallot, H. Schaffers, A. Skarmeta, and S. Decker, "Ecospace - towards an integrated collaboration space for eprofessionals," in *International Conference on Collaborative Computing: Networking, Applications and Worksharing, 2006. CollaborateCom 2006.*, 2006, pp. 1–7.

[8] E. S. Neo, K. Yokoi, S. Kajita, and K. Tanie, "Whole-body motion generation integrating operator's intention and robot's autonomy in controlling humanoid robots," *IEEE Transactions on Robotics*, vol. 23, pp. 763–775, 2007.

[9] N. Mansard, O. Stasse, F. Chaumette, and K. Yokoi, "Visually-guided grasping while walking on a humanoid robot," in *IEEE ICRA*, 2007, pp. 3041–3047.

[10] B. Dariush, M. Gienger, B. Jian, C. Goerick, and K. Fujimura, "Whole body humanoid control from human motion descriptors," in *Robotics and Automation, 2008. ICRA 2008. IEEE International Conference on*, 2008, pp. 2677–2684.

[11] A. Kheddar and A. Escande, "Challenges in contact-support planning for acyclic motion of humanoids and androids," in *39th International Symposium on Robotics*, 2008, pp. 740–745.

[12] O. Stasse, B. Verrelst, P.-B. Wieber, B. V. P. Evrard, A. Kheddar, and K. Yokoi, "Modular architecture for humanoid walking pattern prototyping and experiments," *Advanced Robotics, Special Issue on Middleware for Robotics –Software and Hardware Module in Robotics System*, vol. 22, no. 6, pp. 589–611, 2008.

[13] J. Chesnutt, M. Lau, G. Cheung, J. Kuffner, J. Hodgins, and T. Kanade, "Footstep planning for the honda asimo humanoid," in *IEEE/ICRA*, 2005, pp. 631–636.

[14] J.-M. Bourgeot, N. Cislo, and B. Espiau, "Path-planning and tracking in a 3D complex environment for an anthropomorphic biped robot," in *IEEE/RSJ IROS*, 2002, pp. 2509–2514.

[15] Y. N. W. Takano, K. Yamane, "Capture database through symbolization, recognition and generation of motion patterns," in *IEEE International Conference on Robotics and Automation*, 2007, pp. 3092–3097.

[16] O. C. Jenkins and M. J. Matarić, "Performance-derived behavior vocabularies: Data-driven acqusition of skills from motion," *International Journal of Humanoid Robotics*, vol. 1, no. 2, pp. 237–288, Jun 2004.

[17] E. Drumwright, V. Ng-Thow-Hing, and M. Mataric, "Toward a vocabulary of primitive task programs for humanoid robots," in *International Conference on Development and Learning (ICDL).*

[18] F. Kanehiro, H. Hirukawa, and S. Kajita, "Openhrp: Open architecture humanoid robotics platform," *The International Journal of Robotics Research*, vol. 23, no. 2, pp. 155–165, 2004.

[19] K. Kaneko, F. Kanehiro, S. Kajita, H. Hirukawa, T. Kawasaki, M. Hirata, K. Akachi, and T. Isozumi, "Humanoid robot hrp-2," in *Proceedings of the 2004 IEEE International Conference on Robotics & Automation*, vol. 2, 2004, pp. 1083–1090.

[20] N. Ando, T. Suehiro, K. Kitagaki, and T. Kotoku, "Rt(robot technology)-component and its standardization- towards component based networked robot systems development -," in *SICE-ICASE International Joint Conference 2006 (SICE-ICCAS 2006)*, 2006, pp. 2633–2638.

[21] A. H. Dnni, "The boost statechart library." [Online]. Available: http://www.boost.org/doc/libs/1_34_0/libs/statechart/doc/index.html