[1.] Nikola MALEŠEVIĆ, [2.] Gabor SZIEBIG, [3.] Bjørn SOLVANG, [4.] Tihomir LATINOVIC

SIMULATION OF ROBOTIC TASKS WITH VALIP SYSTEM – PRACTICAL APPLICATION

[1-4.] FACULTY OF MECHANICAL ENGINEERING, BANJA LUKA, BOSNIA & HERZEGOVINA
[2-3.] NARVIK UNIVERSITY COLLEGE, NARVIK, NORWAY

ABSTRACT: In manufacturing engineering, man-machine interaction has gone from typical online programming techniques into virtual reality based offline programming methodologies. Today, a wide range of offline software tools is used to imitate, simulate and control real manufacturing systems. However, these new methodologies lack capability when it comes to human-machine communication. This paper aims to introduce a software platform, called VALIP, where users can build, share and manipulate 3D content, and collaboratively interact with processes in a 3D context, while the participating hardware and software devices can be spatially and/or logically distributed and connected together via the Internet. This unique approach makes the VALIP outstanding among its competitors. The paper also presents a practical solution to a production task.
KEYWORDS: Robot Programming, Virtual Reality, VALIP, Collaboration, Network Communication

INTRODUCTION

As many countries world-wide are starting to implement robotic solutions for manufacturing, educational and other purposes, it is clear that the constant collaboration between the subjects that are experts in this technology is required. In order to expand knowledge, make progress in development of new and existing processes in the field and create cuts in the cost of production and manufacturing where robotics is most heavily used, cooperation between users is mandatory. Reducing the distance between collaborators located world-wide is a major task and most of today's solutions are based on simple Internet communication which is ineffective and can be time-consuming.

This paper tries to take a different approach to the problem and introduce a new way of thinking about this task. By taking advantage of the Internet, advanced applications of information and communication technology and the latest achievements in the software industry, it is possible to make the communication and collaboration more intuitive, effective, and user-friendly.

Continuing from the introduction, in Section 2, we will introduce VALIP system which provides the base for all development and implementation procedures described in this paper. Subsections 2.1 and 2.2 describe the concept of software components (their behaviour and connections) and VirCA (one of the major software tools used for presenting virtual environment to the user) respectively. The practical example which describes a common scenario where VALIP system can be used is presented in Section 3. Lastly, Section 4 contains the conclusion and on overview of the work presented in this paper.

INTRODUCING VALIP

VALIP is an acronym which stands for Virtual Joint Laboratory for Advanced ICT (Information and Communication Technology) in Production. In more readable terms, it represents a commonplace where all collaborators can communicate and exchange information. It represents a virtual copy of the real world and real environment, e.g. laboratory or production environment with machines, robots and other industrial equipment, allowing the collaborators to remotely access the resources that would otherwise be unavailable to them.
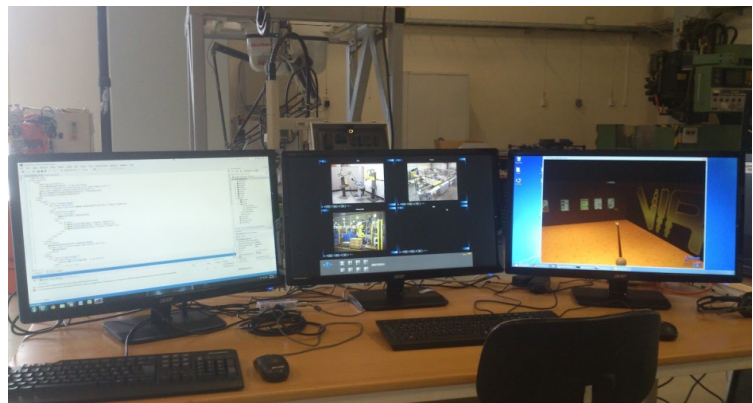

Figure 1. Control workstation with multiple screens, displaying VirCA application window, multiple camera streams and programming environment (from right to left)

VALIP uses a range of tools to accomplish this goal and it consists of software and hardware devices that all communicate together in order to display required information to the participants and allow them to interact with the environment, directly affecting the real world state through virtual workspace.3D visualization can be supplemented by video-camera streaming over the Internet, providing in-depth information and a complete picture about the state of the environment.

Object components

Inside of VALIP environment, every physical or logical object is represented by software component which describes the object and their two most important aspects:

1) The behaviour of the object, taking into account its internal behaviour and the response to actions from other objects connected to it, and
2) The ports or connectors through which the object communicates with other objects in the system.

Since all objects (machines, robots, sensors, logical processes, etc.) in the real world exchange the information between them, it is crucial that their virtual representation has the same ability. This is accomplished by implementing custom ports to software components, based on how they interact with the environment in the real world, through OpenRTM-aist [6], a software platform for development of component-based robotic systems. The components can be developed in several programming languages, most notably C++ and Python.

The communication between the components is established through one of the following port types, which can be based on either CORBA or ICE implementation [1]:

1) Data Port – used for continuous exchange of information between the components (CORBA based),
2) Service Port – used for interface-based communication defined by IDL (Interface Description File) (CORBA based),and
3) Ice Port – used for interface-based communication defined by Slice (Specification Language for Ice) port (Ice based).

In addition, each port can either act exclusively as a provider or a consumer. This means that in order for components to have a two-way communication between them, each of them has to have a set of provider/consumer ports. The example for component layout and their connections will be presented in section 3.

Virtual environment

VALIP uses software package VirCA as its primary tool for presenting the information to the user [2] [7]. VirCA is an acronym and stands for Virtual Collaboration Arena. It is a software application which renders a 3D environment inside of which virtual reality can be presented to end users. Collaborators can interact with the objects displayed in application window and inspect the current state of the virtual world (which, as already mentioned, represents the copy of the real world) in real-time [3].

The content inside of virtual workspace is defined by components that are connected to VirCA component. These components constantly send information about their virtual representation allowing VirCA to render their 3D models or other graphical details inside of virtual world.

The user is able to interact with the components through VirCA application window or by using dedicated control components. Sample tasks include turning on the conveyer belt or instructing the robot to pick up an object.

PRACTICAL EXAMPLE

Practical example of VALIP-based system is demonstrated by performing a remote operation on robotic cell located at Narvik University College (NUC). The task described represents a basic functionality of the robotic cell and contains procedures which are implemented in any other task robots commonly perform in industrial environments. Almost all other types of robotic assignments are constructed upon simple instructions of robot movement, gripper actuation (analogously welding control) and sensor reading, all of which are included in the steps of this task.

For the sake of simplicity, the cell consists of a single robot with the gripper attached to it, one conveyer belt equipped with sensors and a simple table. The robot task consists of following steps:

☐ picking up the object from specific position on the table,
☐ placing the object on the conveyer belt,
☐ waiting for the object to trigger one of the detection sensors located on conveyer belt,
☐ positioning the robot at object pickup location,
☐ waiting for the object to approach the position,
☐ picking up the object,
☐ placing the object back at the original pickup position on the table.

The system consists of multiple software components which are run on three computers. Server computer is located at NUC and provides a nameserver to which all of the components connect and communicate through. Robot computer is also located at NUC and is directly connected to KUKA robot. Finally, client computer is located at UBL and is used to display virtual reality to remote users. Each of the components is described below.
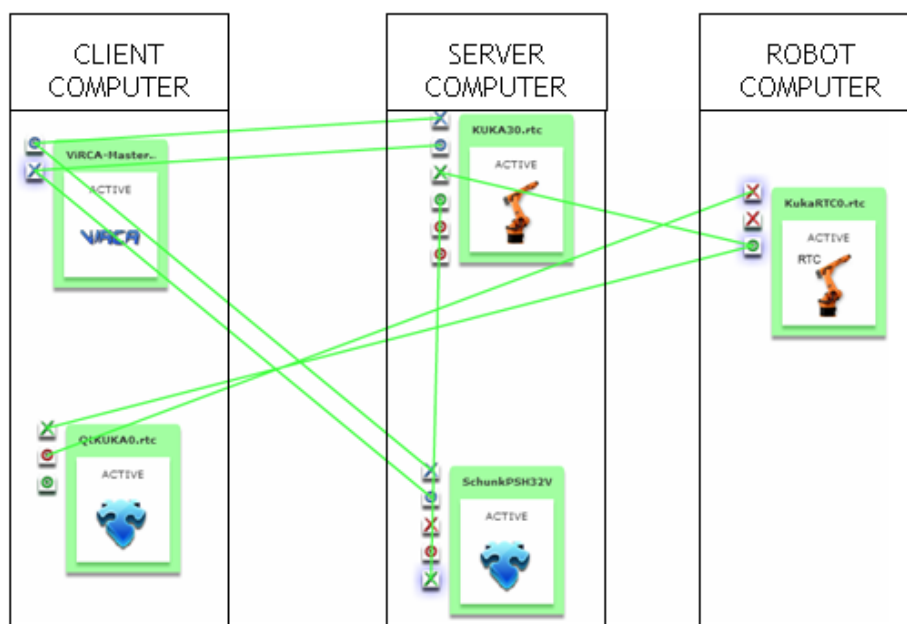
Figure 2. Component layout and connections between them

1) **KukaRTC.** Used for robot control, this component is run on a dedicated robot computer. This computer is directly connected to the robot and also connected to the server computer via separate LAN network or Internet connection. The components allows VALIP to communicate to robot and exchange data with it.

2) **Kuka3.** Used for providing VirCA with the information about robot position and orientation, allowing it to render robot model. This component is run on server computer.

3) **SchunkPSH32Vis.** Used for providing VirCA with the information about gripper position and orientation, allowing it to render gripper model. This component is run on server computer. It is designed as a separate component to allow gripper to be easily replaced with different model (with its own visualization component).

4) **QtKUKA.** Used to provide additional GUI to allow further control by the user. While it is also possible to control the task directly through VirCA, this GUI presents more advanced options to the user, making the whole user experience more intuitive. This component is run at client computer, allowing for remote operation.

5) **VirCA.** Software tool used to render the environment and component models as a virtual reality. This component is run at client computer, allowing for remote operation.

All components are connected to each other, exchanging the information between them. The connections are established using System Editor, yet another software tool, located at server computer. Ice ports are marked in blue colour and are mainly used for visual information exchange between visualization components and VirCA. CORBA Data ports are marked in green colour and are used for continuous feed of information between components. Red coloured CORBA data ports are used for IDL-based control interfaces.
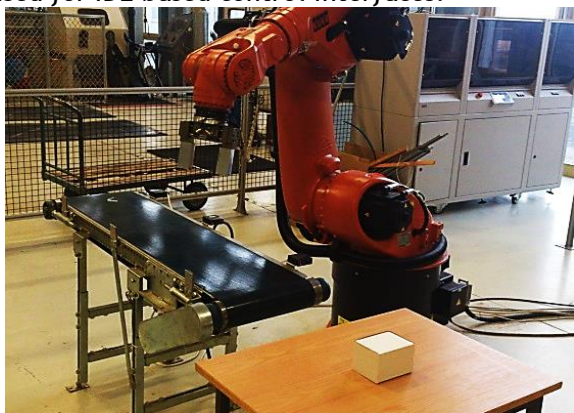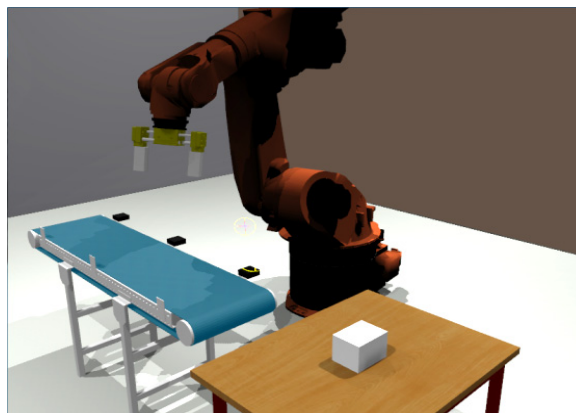

Figure 3. Physical robotic cell   *Figure 4. Robotic cell in virtual reality as rendered by VirCA*

Since the robot is already programmed for this task, the process is being monitored through VirCA from Laboratory for Intelligent Systems at University of Banja Luka (UBL) which is equiped with required hardware and software. More importantly, the whole task can be manually started, positions and other parameters modified and process controled from this location, more than 2.600 km away from the physical robotic cell (or any other location world-wide, for that matter).

CONCLUSIONS

Conventional solutions for remote monitoring (e.g. camera streaming) and control (e.g. web interface) of environments such as robotic cells are appropriate solutions for simpler configurations. However, when high complexity of the system is brought into the equation, these methods slowly become obsolete and new ways of problem solving has to be implemented in order to enact a full control over the environment[4] [5].

There are numerous applications for VALIP, some of them being remote control of production lines for companies that have facilities spread in world-wide locations, or teaching of robot programming methods to students/workers without the need of having physical equipment at their location.

This paper was focused on describing VALIP as a feasible solution that fills in the gaps that are present in conventional methods of control and monitoring. VALIP represents highly modular and customizable product, and offers tools that allow end users to easily integrate components of robotic cells, production lines and other similar environments, allowing personnel to remotely interact with them. In this process, users are offered more intuitive, sensible and understandable overview of the whole system.

ACKNOWLEDGMENTS

REFERENCES

[1] Houston P.J., Wilkie F.G., Anderson T.J. (1998). Component-based development, CORBA and RM-ODP. IEE Proceedings – Software, Vol. 145, No. 1, p. 22-28

[2] Galambos P., Baranyi P. (2011). VirCA as Virtual Intelligent Space for RT-Middleware. 2011 IEEE/ASME International Conference on Advanced Intelligent Mechatronics. p. 140-145

[3] Sziebig G., Øritsland, T.A. (2012). Navigating in 3D Immersive Environments: a VirCA usability study. 10th IFAC Symposium on Robot Control International Federation of Automatic Control. p. 380.-384

[4] Solvang B., Sziebig G. (2012). On Industrial Robots and Cognitive Info-Communication. 3rd IEEE International Conference on Cognitive Infocommunications. p. 459-464

[5] Aryania A., Daniel B., Thomessen T., Sziebig G. (2012). New trends in industrial robot controller user interfaces. 3rd IEEE International Conference on Cognitive Infocommunications. p. 365-368

[6] National Institute of Advanced Industrial Science and Technology. OpenRTM-aist, from http://www.openrtm.org/openrtm/en/node/629, accessed on 2013-04-09

[7] Joint Laboratory for Emerging Information Technology in Manufacturing (ITM). VirCA, from http://www.virca.hu, accessed on 2013-04-09

ANNALS of Faculty Engineering Hunedoara

– International Journal of Engineering