Component Based Systems out of "Virtual" and "Real" Bricks

Nikos Drossos, Manolis Koutlis
R&D Engineer – Senior R&D Engineer/Researcher
Computer Technology Institute
Patras, Greece
ndrossos@cti.gr, koutlis@cti.gr

SUMMARY

This paper presents a system that was designed and developed to integrate E-Slate, a component-based environment for building educational software with the Lego Mindstorms™ system. The system allows for the construction of hybrid educational "Microworlds" consisting of both software entities and real-world devices (Lego bricks, motors, etc), providing in that way the means for combining mathematical modeling with kinesthetic experience in a synchronized way. The systems' design clearly demonstrates how the metaphors of component-oriented design from software engineering can be smoothly integrated with the domain of tangible devices. Although initially targeting the domain of educational applications the design concepts can be applied in many other domains.

KEYWORDS: *Component based software, hybrid systems, robotics, systems integration*

INTRODUCTION

This paper reports on accomplished research work concerning the design and development of a hybrid component based system, encompassing both hardware and software components, which provides a platform for the construction of a number of applications.

The system builds upon two existing products and integrates them smoothly, augmenting each one with the best of the two worlds. Specifically, the system described here uses the educational environment "E-Slate" (http://E_Slate.cti.gr), a research product developed at the Computer Technology Institute of Greece, and the Lego Mindstorms™ system, based on devices initially implemented in the Artificial Intelligence Laboratory of MIT for constructing educational activities.

E-Slate

E-Slate is an exploratory learning environment that builds on a component-based approach of software development to fulfill some core requirements stemming from the educational domain, namely end-user authoring and support for the "learning by doing" approach.

Lego Mindstorms

Lego Mindstorms are reduced size computers, able to receive input from electrical sensors, for example touch and light, and to manipulate several actuators like motors (their development started at the middle 80's, within the framework of a research project of Media Laboratory of MIT, and later on, extended by the Lego Group of Denmark and became a commercial product).

SYSTEM DESIGN

Within the scope of this research, an integrated system is designed and developed realizing the means for interconnecting real-world devices with software counterparts, by end-users. The prototype that was developed comprises a kind of a "glue" system that interconnects E-Slate's software components with the Lego Mindstorms. Specifically:

- *It provides the means for E-Slate's components to collaborate in real time with the Lego devices.*

RCX programming now changes and enhances interactivity. The RCX is now capable of executing programs not only for specifying functionalities for the RCX itself, but also programs that can take input, respond to actions, perform operations through the RCX's actuators and send feedback to the computer.

- *It allows multiple Lego units to participate in a Microworld (cluster of interconnected hardware and software components), by inventing name addressing of Lego devices inside E-Slate.*

The system can now control many different RCX units and have them cooperating with the E-Slate corresponding components.

- *Allows bi-directional communication between E-Slate and Lego Mindstorms.*

E-Slate, on the one hand, can send either data or plain notifications, instructing the RCX to execute operations or manipulate its actuators in a certain way, and, on the other hand, RCX can notify or inform E-Slate about the input it receives through its sensors.

- *Provides scripting language support to the Lego Mindstorms.*

Apart from manipulation of the system through the graphical user interface, one can also interact with it through scripting languages. Having E-Slate supporting Logo, JavaScript and Java, things are much more feasible.

System Architecture

The system consists of a number of hardware RCX devices on the one side, a computer running E-Slate on the other and an infrared tower between them in order to facilitate communication.

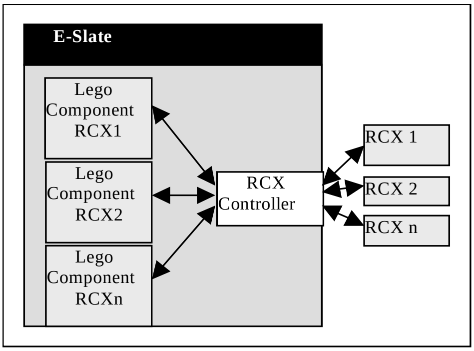

Figure 1: System Architecture

For every RCX in the system there is a corresponding Lego component inside E-Slate called "LegoComponent" that has the same name as the real device. Thus, every component inside E-Slate is a software representation of the hardware device.

The existence of many Lego Components and RCXs implies the need of an extra controller component called “RCXController”, to facilitate the communication among the hardware and the software components. This software component is responsible for the management of the serial port of the computer and the routing of data from the sender to the correct receiver and vice versa. As it is shown in the above picture, communication between Lego Components and RCX Controller as well as between the RCX Controller and the real devices is bi-directional.

APPLICATION EXAMPLES

The described system can very easily apply to many different application domains, using a powerful combination of software and hardware components to illustrate in a tangible way the various notions of each one.

Lego enhanced E-Slate microworld

The first microworld utilizes simple E-Slate components to illustrate the programming features through the Logo scripting language. A snapshot of the microworld is showing below.

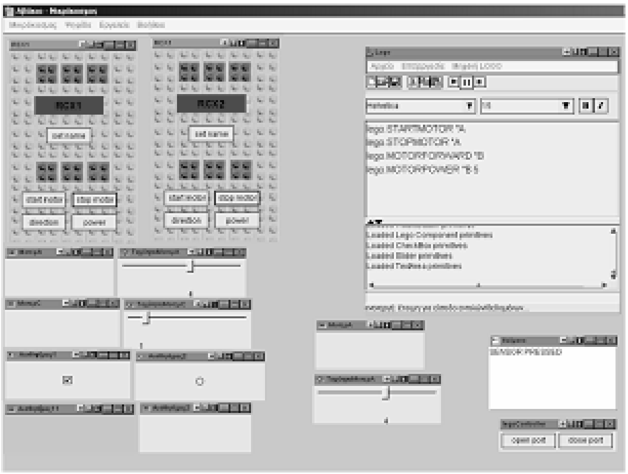

Figure 2: Two-lego microworld

There are two Lego components each one connected to buttons and sliders controlling the state and the speed of the motors, and other buttons to reflect the state of the Lego’s sensors. There is also a text component used in the event scripting mechanism demonstration and finally RCX Controller.

USING THE SYSTEM

The initialization and interaction of the system can be summarized in the following steps:
Initial Step 1: After the initialization of the E-Slate microworld, the communication initialization between the serial port and the infrared tower has to take place in order to enable the interaction with the RCXs.
Initial Step 2: The second step for the initialization phase is the naming of all the RCX devices participating in the microworld. This step can be done by using the graphical user interface of the Lego component or by using the logo instruction lego.SETNAME.

After the above initialization is repeated for every RCX in the system, the microworld is then ready to interact with the real devices.
There are five different ways to interact with the system:

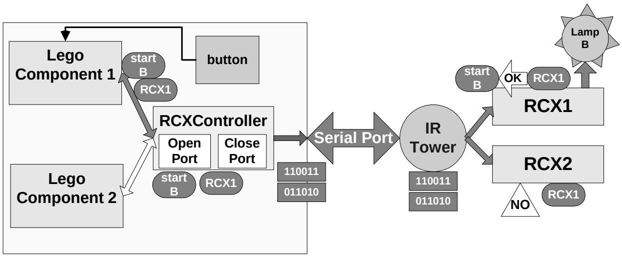

Figure 6: Interacting with the system

1st way: By using the graphical user interface of the Lego component.
2nd way: By using the other e-slate components connected through plugs to the Lego component.
3rd way: By using the logo component and the supported instruction set for the Lego component.
4th way: When the state of a sensor changes, all the components plugged to the Lego component are notified. For example if the touch sensor is pushed, and there is a button plugged to the corresponding plug of the Lego component, the state of the button will change to pressed.
5th way: In this case the change of a sensor's state fires an event that causes the execution of a script.

CONTRIBUTION

The contribution of this work is multifold:

- It provides the means and tools for the construction of a multitude of educational activities that combine mathematical modeling with kinesthetic experience in a synchronized way.
- It extends the metaphor of component-oriented system design from software engineering to the domain of tangible devices by smoothly integrating concepts of both worlds.
- It augments the Lego Mindstorms system (or any other similar) with the ability to connect to software entities within a "virtual" domain.
- It augments the E-Slate educational platform with the ability to control a multitude of real-world devices.

REFERENCES

Bruce Eckel (2000), "*Thinking in Java Second Edition*", Prentice Hall, 2000
G. Birbilis, M. Koutlis, K. Kyrimis, G. Tsironis, G. Vasiliou, (2000): "E-Slate: A software architectural style for end-user programming", presented at the 22nd International Conference on *Software Engineering*, Limerick, Ireland.
Lego Mindstorms Internals. URL: http://www.crynwr.com/lego-robotics/
leJOS Homepage. URL: http://lejos.sourceforge.net.
Manolis Koutlis (2001), "Computational Environments for Teaching and Learning in schools, with application in Geography". PhD dissertation, University of Patras, Department of Software Engineering and Informatics.
MIT Media Laboratory (1997), "*The Programmable Brick Handbook*".
Tom Spitzer (1997), "*Component Architectures*", DBMS September 1997.