



### OPEN COMPONENTS FOR EMBEDDED REAL-TIME APPLICATIONS

# Robotic Application demo CEA

## **Goals:**

Robotic application provides a validation tool for Ocera components. It consists in a porting of an existing haptic robot device controller from the VxWorks RTOS to Ocera Linux/RTLinux. Initial release of the application (V1) operates on a simulated robot. It is not intended to provide a "realistic" behavior and does not feature all Ocera components that will be included in the final application (V2).

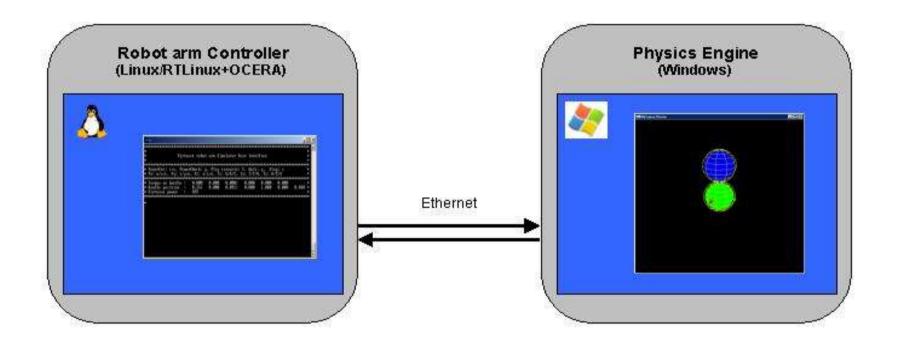
## **Tested components:**

- Ocera Linux/RTLinux executive
- Dynamic memory allocation
- POSIX Trace
- POSIX Barriers
- Fault Tolerance
- RT-UDP

## **Description:**

The Ocera Robotic application is an embedded controller for an haptic robot device named Virtuose. The control software performs bi-lateral coupling in force and position between the Virtuose robot arm and a virtual object dynamically simulated on a remote computer running a Physics Engine. The controller and the remote computer are connected through Ethernet.

Initial release of the application (V1) does not perform control of a real Virtuose robot but of a simulated one instead. User can interact with the simulated Virtuose robot using a plain text interface on the Controller and with the virtual object using a GUI interface on the remote machine. The goal of application V1 is not to provide a realistic behavior: only high level functional behavior, such as communications, mode switches and coupling controls are tested in the demo.



Any efforts applied on the simulated Virtuose robot handle (through the Controller user interface) induces motion of both the Virtuose handle and the green ball (mobile coupled virtual object). Any effort applied on the green ball (through the Physics Engine GUI) induces motion of both the green ball and the Virtuose handle. When a collision occurs between the green ball (mobile coupled virtual object) and the blue ball (fixed virtual object) within the Physics Engine, force feedback repulses the Virtuose robot handle in the opposite direction.

## **Demo phases**

#### Phase 1

Start the overall Robotic application:

- Start the Robot arm control application on the Controller using the provided script (run). Check for correct initialization (text message displayed on the Controller console).
- Start the Physics Engine on the remote computer. Check for correct initialization and synchronization with the Controller (by applying efforts on green ball to check for elastic link to fixed position).

#### Phase 2

Control coupling of the Virtuose handle and the green ball:

- From the Virtuose user interface, power on the robot and apply efforts on the Virtuose handle. Check for motion of both the handle and the green ball.
- From the Physics Engine GUI, apply efforts on the green ball and check for motion of both the green ball and the Virtuose handle.
- From the Virtuose user interface, move the Virtuose handle in order to collide the green ball with the blue ball. Check for handle repulsion.

#### Phase 3

Play the test scenario. The test scenario simulates a predefined user motion and stores to a file force applied on and position of the Virtuose handle. The scenario is designed so that a collision between the green ball and the blue ball occurs during the motion along a straight line path.

- From the Virtuose user interface, start the test scenario and wait for completion.
- Plot the generated output file using gnuplot with the provided gnuplot script. Check for correct collision behavior.

## **Conclusions and results:**

Application V1 demo execution shows the expected behavior of the Haptic control software performing bi-lateral coupling between the simulated Virtuose robot arm and a mobile virtual object (green ball) dynamically simulated on a remote computer running a Physics Engine (currently a simplified version). Because of some technical issues, not all expected Ocera features have been implemented in this application release. The missing features will though be implemented in the final release (V2) that will work with a real Virtuose robot arm and a full featured Physics Engine application (using Vortex physics engine).