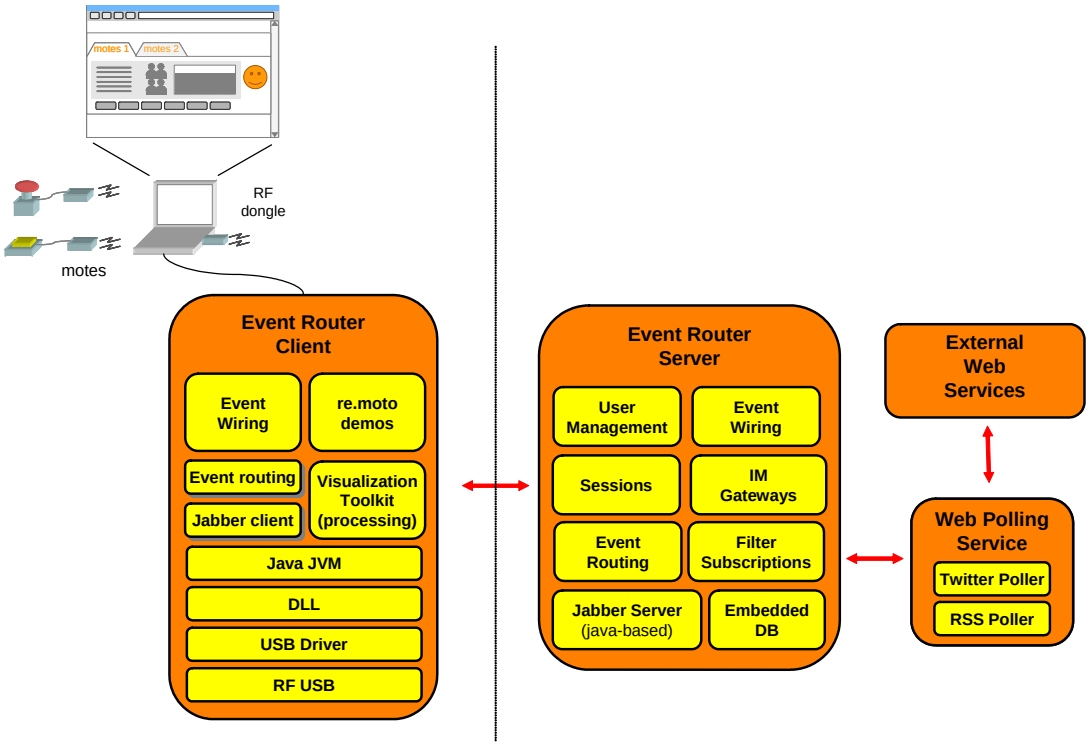
Re.moto Server and Client

Introduction

The re.moto system is an Internet-wide event notification system for sensors and devices. Users connect their devices or sensors to their PCs by way of RF sensors and the events generated by the sensors get sent to the re.moto server. Other devices, or clients can subscribe to these events (or filter the public stream) and receive those events, which they can visualize, be sent to their IM accounts, or perform some additional scripting.

The initial implementation of the system is geared to providing a PC-client that can connect the devices as well as visualize the events in various ways. The PC-client uses various Widgets to display the outputs/effectors of the sensors/devices as well as enable the interconnections among the various components emitting the events.

Architecture



The diagram above reflects the architecture of the initial implementation of the system.

Event Routing

The router will be based on a Jabber platform. That is, the server will be a Jabber server where Jabber clients will connect to and send XMPP packets to be routed to the appropriate destination. In the PC's where the sensors will be connected to, there will be a driver that will listen to sensor events and via a Jabber client, it will send those events to the Event Router server. Other-devices subscribed to the service, will be able to setup filters (keyword-based, topics, sensor-ids) to select those events that they would be interested on and send it when the user/device would be connected. In order to visualize the events, a user would setup visualization widgets (initially part of the PC-client application) that would subscribe to those events interested in visualizing and the Event Router server would deliver them much like any other device. In effect, the visualization widgets are seen by the Event Router server as virtual devices.

Event wiring

As explained in the section above, sensors would emit events, which other devices/effectors can subscribe to. We refer as wiring, the setup to connect the events of one sensor into the input of another device/effector. Wiring a device can specify the events of either a particular sensor/device, or that of many in a topic, or that of many in a keyword-based filter on the event packets. A particular set of wires would be packaged as a layout. The PC client

may have a visualization software to show the layout and if the devices are virtual (visualization widgets) and local to the PC, then these will be shown as part of the layout.