Per Application virtual keyboard and application launcher as part of a Multi-Touch, Multi-User Linux Distribution

GSoC 2012 Proposal

Vladimir Shehov

The project goal is to make possible big Multi-Touch devices (like Tables or Walls) to be used by many people, located at different places on the device at the same time. To achieve this goal, there must be a suitable launcher to launch applications from every side of the device. Each application must support rotation and have its own virtual keyboard.

Introduction

My name is Vladimir Shehov. My email is vladimir.shehov@mail.bg . I'm 24 years old and currently studying for Masters Degree, specialty "Electrical Power Engineering and Electrical Equipment" at "Technical University of Sofia", Bulgaria (GMT +2). My Bachelors Degree is on same specialty. Recently my hobbies are the MT Technologies. I'm writing my Master Thesis, named "Multitouch tables by using IR technologies". I have almost finished my DSI+LCD based table. I have many years working experience in programming and system administration. More than 8 years of C programming, recently and some C# programming .

I would like to write few words about the reason that I think this project is going to be useful. I want to make complete software solution, for example linux distribution made for MT environment, that you can download it, install it easily and start working with it. I decided to be linux mainly because most of the software is open source and it is easy to be modified. So I started thinking what I need to make touch screen linux distribution that can be used by many people simultaneously:

 - Firstly the applications must work without modifications or with smaller ones, this way the existing applications like browsers, office packets and etc. can be used.

- Many people should work on the table and they may be located on different sides of the table, so I must use some composite window manager to support rotation of the applications.
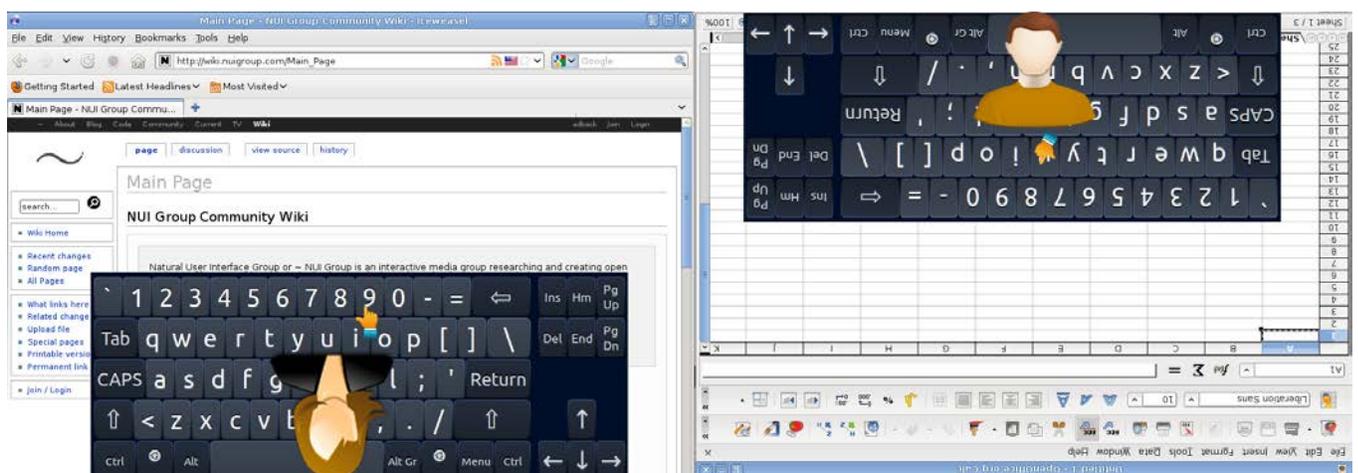
- Every application must have its own keyboard, because 2 or 3 persons at a time may use it, for example 2 browsers on different sides of the table.

- There should be suitable application launcher made for MT environment. For example when somebody clicks with 3 fingers in one place for 1 second, the launcher may appear on that place.

For my goals I'm going to use Compiz or some other composite window manager and going to modify TUIO for MPX X Server (which was produced in GSoC 2009) to use latest X input extensions (http://www.x.org/wiki/XI2).

I will try to provide not only "virtual keyboard per application" and application launcher, but fixes and improvements for some existing applications to use newest X input extensions. Also I will try to provide HOWTOs for the things that I face problems or I think that could be useful to someone else.

Project description

Per application virtual keyboard

The main objective of the project proposal is to create virtual keyboard that sends input only to the application, which invoked it. This way every application is going to have its own keyboard. Many users will be able to interact with different applications on the MT device at the same time.

It is important that the keyboard can be used with any touch device with existing input driver for X. For this reason I would like to use latest X input extensions. This way the keyboard can support not only TUIO based touch devices, but also devices with touch frames that use their own input driver.

For the interface part I may use GTK+ 3.0 or 3.2 because it supports XInput2 and it is widely used and supported. The interface is going to be simple. Every application is going to have some kind of keyboard icon placed on the top-level window on the title bar or somewhere else. When the user clicks it, a keyboard will appear. After clicking the button again, the keyboards will disappear. If everything is going smoothly and I have accomplished my priority tasks I can make different keyboards that depend on the size of the top-level application window. If the window is smaller, the keyboard can display only letters (like phones are). In such case the numbers and the symbols will appear after clicking a button.

MT Application Launcher

My second task is creating an application launcher for multi-user environment. The main reason is to make launching of the applications by the people located on different sides of the device easier. To make it possible I think to use some gesture mechanism like clicking with 3 fingers and holding them on the device for 1 second. Then the launcher will appear. The launcher will provide the option to group the applications in categories.

The launcher will use easily editable xml based configuration for the applications. It will contain:

1. Common configuration for the launcher.

2. Application groups (categories) on higher level.

3. On lower level are the applications.

Example configuration File:

```
<Config TouchPoints="3" Delay="1000" />
<AppGroup Name="Internet" Description="Internet Applications"
Icon="/usr/share/images/internet.png">
        <Application Name="Chrome" Description="Chrome web browser"
Icon="/usr/share/images/chrome.png" Path="/usr/bin/chrome" ExecutionLimit="" />
        <Application Name="Skype" Description="Skype client"
Icon="/usr/share/images/skype.png" Path="/usr/bin/skype" ExecutionLimit="" />
</AppGroup>
<AppGroup Name="Office" Description="Office Applications"
Icon="/usr/share/images/office.png">
        <Application Name="LibreOffice" Description="" Icon="" Path="/usr/bin/libre"
ExecutionLimit="1" />
</AppGroup>
```

For parsing of the configuration I may use libxml. Also I may need to make some improvements on the current TUIO input driver to use newest features of XI2.2

http://www.phoronix.com/scan.php?page=news_item&px=MTAyODc

Timeline

April 24 – May 21 (Before the official coding starts)

I will use this time to get more familiar with XI2.2. I will also use the time to read more documentation about other X libraries that can be useful. I'm going to check some of the sources of current TUIO input driver and make plans of adding additional features that provides XI2.2.

May 21 – July 9 (The official coding time to the mid-term)

I will write some examples using the newest X input extensions and other libs that I will use in my projects. I will start making tests with the application events. After that I will create a simple test application that sends different input to the applications. At the end I will start making the interface.

July 9 – August 9 (After the mid-term)

Firstly I'm planning to finish with the virtual keyboard and make some tests. Then I will start working on the launcher.

August 9 – August 20 (The pencil down date)

Time left for some improvements or unpredicted delay.

Deliverables

Documentation/Examples/Notes:

During the process of writing the code and reviewing the X libraries I'll write examples of using XI2.2 for multi-touch applications. It might be helpful if anybody is interested in writing multi-touch applications with the newest X input extensions.

Per application virtual keyboard. Before the middle term evaluation:

Before the middle term evaluation I'm planning to have significant progress on sending different input to the applications. Also I'll write some tests for that. I will start making the interface of the keyboard.

Per application virtual keyboard. After the middle term evaluation:

My plans are to finish the interface of the keyboard and to test it. Also I'll write some documentation how can be integrated.

MT Application Launcher. By the end of GSoC:

By the end of GSoC I'm planning to finish the launcher. I plan to make some tests on it and to write documentation and practical configuration examples for using it.