Handling pointer events in Symbian C++

Overview

S60 5th Edition supports touch events. All AVKON UI controls handle touch UI events and you do not need to implement anything new.

However, in your custom CCoeControl UI control you must implement event handling yourself. This can be done by implementing the virtual base class method `CCoeControl::HandlePointerEventL()`.

The most common touch events are:

```
* EButton1Down (=pen down)
* EButton1Up (=pen up)
* EDrag (dragging with pen)
```

The other events can be found from struct TPointerEvent in the w32sth.h header.

MMP file

The following libraries are required:

```
LIBRARY cone.lib
```

Header file

```
private:
    void HandlePointerEventL(const TPointerEvent& aPointerEvent);
```

Source file

It is good practice to call the base class method from your custom control `HandlePointerEventL()` method. If your custom control is a container control that has child CCoeControls, then they will receive pointer events too, which is what the base class implementation does.

```
void CMyContainer::HandlePointerEventL(const TPointerEvent& aPointerEvent)
    {
    // Remember to call base class implementation. Then your child controls
    // receive pointer events.
    CCoeControl::HandlePointerEventL(aPointerEvent);

    // Rest of your code
    if (aPointerEvent.iType == TPointerEvent::EButton1Down)
        {
        // TODO: What to do when this ui control receives pen down event...
        }
    }
```

Postconditions

Touch event received.

(C) Copyright Nokia 2014. All rights reserved.

Supplementary material

- You can test pointer events handling in a simple, executable application into which this code snippet has been patched. The application is available for download at: Media:ExampleStub w pointer events.zip
- You can examine all the changes that are required to implement pointer events handling in an application. The changes are provided in unified diff and color-coded diff formats: Media:CS001144 Handling pointer events.diff.zip
- For general information on applying the patch, see Using Diffs.
- For unpatched stub applications, see Example stub.

See also

- S60 5th Edition: Solitaire Game Example
- S60 Platform: ImageConverter Example
- Pointer capturing in Symbian C++ on touch-enabled devices
- Enabling drag events in Symbian C++ on touch-enabled devices

(C) Copyright Nokia 2014. All rights reserved.