How to force launch of VKB in FEP-aware custom control

Overview

Implementation of the FEP-aware custom text control is quite straightforward. The main idea is to implement MCoeFepAwareTextEditor_Extension1 interface, which stores control's state used by FEP, and MCoeFepAwareTextEditor interface which specifies a protocol for FEP-aware text editors (and mainly responsible for the inline-editing functionality). A detailed description of the mentioned interfaces is shown in the S60 SDK documentation for both 3rd and 5th Edition. But with the introduction of touch support on S60 5th Edition the problem of launching virtual keyboard (VKB) in custom FEP-aware custom control appeared.

Solution

The solution of the specified problem lies in the usage of new values added to **MAknEdStateObserver::EAknEdwinStateEvent** enumeration:

on S60 3rd Edition

```
class MAknEdStateObserver
         {
public:
enum EAknEdwinStateEvent
         {
         EAknEdwinStateEventStateUpdate,
         EAknEdwinStateInputModeUpdate,
         EAknEdwinStateCaseModeUpdate,
         EAknEdwinStateLocalLanguageUpdate,
         EAknEdwinStateFlagsUpdate
         };
...
};
```

and on S60 5th Edition

```
class MAknEdStateObserver
         {
public:
enum EAknEdwinStateEvent
         {
         EAknEdwinStateEventStateUpdate,
         EAknEdwinStateInputModeUpdate,
         EAknEdwinStateCaseModeUpdate,
         EAknEdwinStateLocalLanguageUpdate,
         EAknEdwinStateFlagsUpdate,
         EAknSyncEdwinState,
         EAknCursorPositionChanged,
         EAknActivatePenInputRequest
         };
...
};
```

As it can be seen, three new items have been added on S60 5th Edition. So to force launch of the VKB in FEP-aware custom control we should make the following call (e.g. in control's HandlePointerEventL() method):

(C) Copyright Nokia 2014. All rights reserved.

```
void CCustomTextEditor::HandlePointerEventL(const TPointerEvent &aPointerEvent)
{
...
  CAknEdwinState* editorState = (CAknEdwinState*) iFepExtension->State(iTextFepUid);

  editorState->ReportAknEdStateEventL(MAknEdStateObserver::EAknActivatePenInputRequest);
...
}
```

Here `iFepExtension` is the pointer to our implementation of the `MCoeFepAwareTextEditor_Extension1` interface, `iTextFepUid` is the uid of the FEP which was associated with our control while constructing iFepExtension. However `iTextFepUid` could have custom value. The only need is to store this value somewhere in our custom control.

Postconditions

Launch of the VKB could be forced in FEP-aware custom control.

(C) Copyright Nokia 2014. All rights reserved.