
Subject: Re: FEATURE_REQUEST: SliderCtrl to have WhenClick

Posted by [mrjt](#) on Thu, 22 Oct 2009 13:58:51 GMT

[View Forum Message](#) <> [Reply to Message](#)

I can't comment on why Ctrl don't have more default actions, but everything you want can be done easily using a composite control:

```
class KeySlider : public Ctrl
```

```
{
```

```
    typedef KeySlider CLASSNAME;
```

```
private:
```

```
    struct ClickSlider : public SliderCtrl {
```

```
        Callback WhenClick;
```

```
        virtual void LeftUp(Point p, dword keyflags) {
```

```
            SliderCtrl::LeftUp(p, keyflags);
```

```
            WhenClick();
```

```
        }
```

```
};
```

```
ClickSlider ctrl;
```

```
Label label;
```

```
Vector<Value> keys;
```

```
void UpdateSlider();
```

```
void Click() const { WhenClick(); }
```

```
virtual void Updated() { label.SetLabel(AsString(GetData())); }
```

```
public:
```

```
    Callback WhenClick;
```

```
KeySlider();
```

```
virtual void LeftUp(Point p, dword keyflags) { Click(); }
```

```
KeySlider &Add(Value v) { keys.Add(v); UpdateSlider(); return *this; }
```

```
void Remove(int i) { keys.Remove(i); UpdateSlider(); }
```

```
int GetCount() const;
```

```
void SetLabelWidth(int cx);
```

```
virtual Value GetData() const { return keys.GetCount() ? keys[(int)~ctrl] : Value(); }
```

```
virtual void SetData(const Value& data);
```

```
};
```

```
KeySlider::KeySlider()
```

```
{
```

```
    Transparent(true);
```

```
    SetLabelWidth(32);
```

```
    UpdateSlider();
```

```
    ctrl.WhenClick = THISBACK(Click);
```

```
    ctrl <<= THISBACK(UpdateAction);
```

```

}

void KeySlider::SetLabelWidth(int cx)
{
    Ctrl::Add(ctrl.VSizePosZ().HSizePosZ(0, cx+2));
    Ctrl::Add(label.VSizePosZ().RightPosZ(0, cx));
}

void KeySlider::UpdateSlider()
{
    ctrl.Enable(keys.GetCount());
    if (!keys.GetCount())
        return;
    ctrl.MinMax(0, keys.GetCount()-1);
    if (IsNull(~ctrl))
        ctrl <= 0;
    Updated();
}

void KeySlider::SetData(const Value& data)
{
    int ix = FindIndex(keys, data);
    if (ix >= 0)
        ctrl <= ix;
}

```

Because it uses key values you can do things like:

```
slider.Add("A").Add("B").Add("C").Add("D").Add("E").Add("F").Add("G");
```

And when I want to add WhenClick or WhenKey events to Ctrl's I usually just use a simple template that overloads the event handling function.