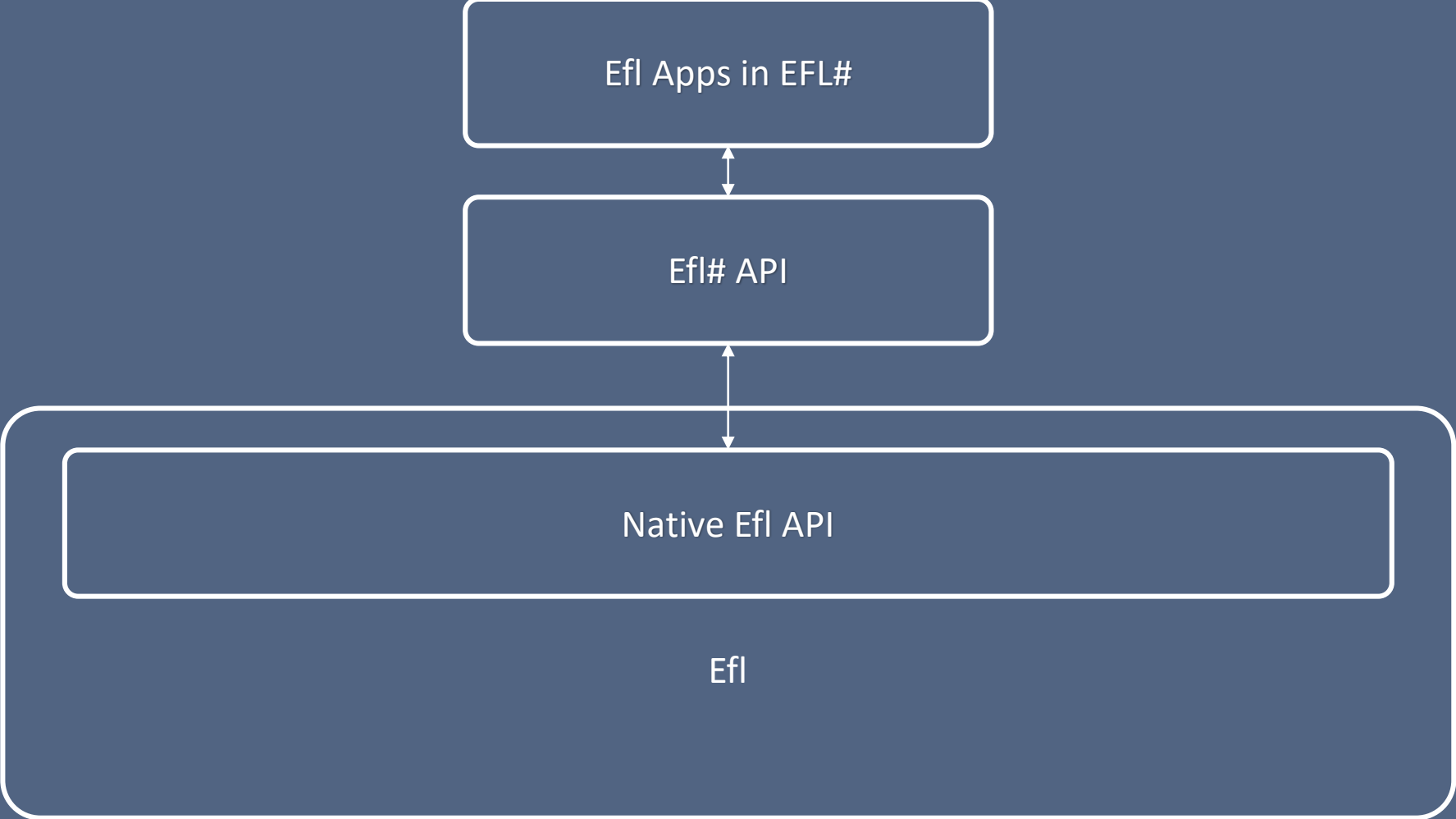


EFL# Binding and Sharing Development Experience

Taehyub Kim | herb
Taehyub.kim@samsung.com

C# Binding for EFL

- Eo files
- Eolian_mono generator (...cs files)
- efl_mono.dll



Native EFL vs EFL#

EFL# AlertPopup

```
Efl.Ui.AlertPopup popup = new Efl.Ui.AlertPopup(win);
popup.TitlePart.SetText("Title");
popup.SetHintSizeMin(new Eina.Size2D(200, 200));
popup.SetScrollableText("This is text");
popup.SetButton(Efl.Ui.AlertPopupButton.Positive, "Ok", null);
popup.SetButton(Efl.Ui.AlertPopupButton.User, "Cancel", null);
popup.ButtonClickedEvent += Popup_ButtonClickedEvent;
```

Native AlertPopup

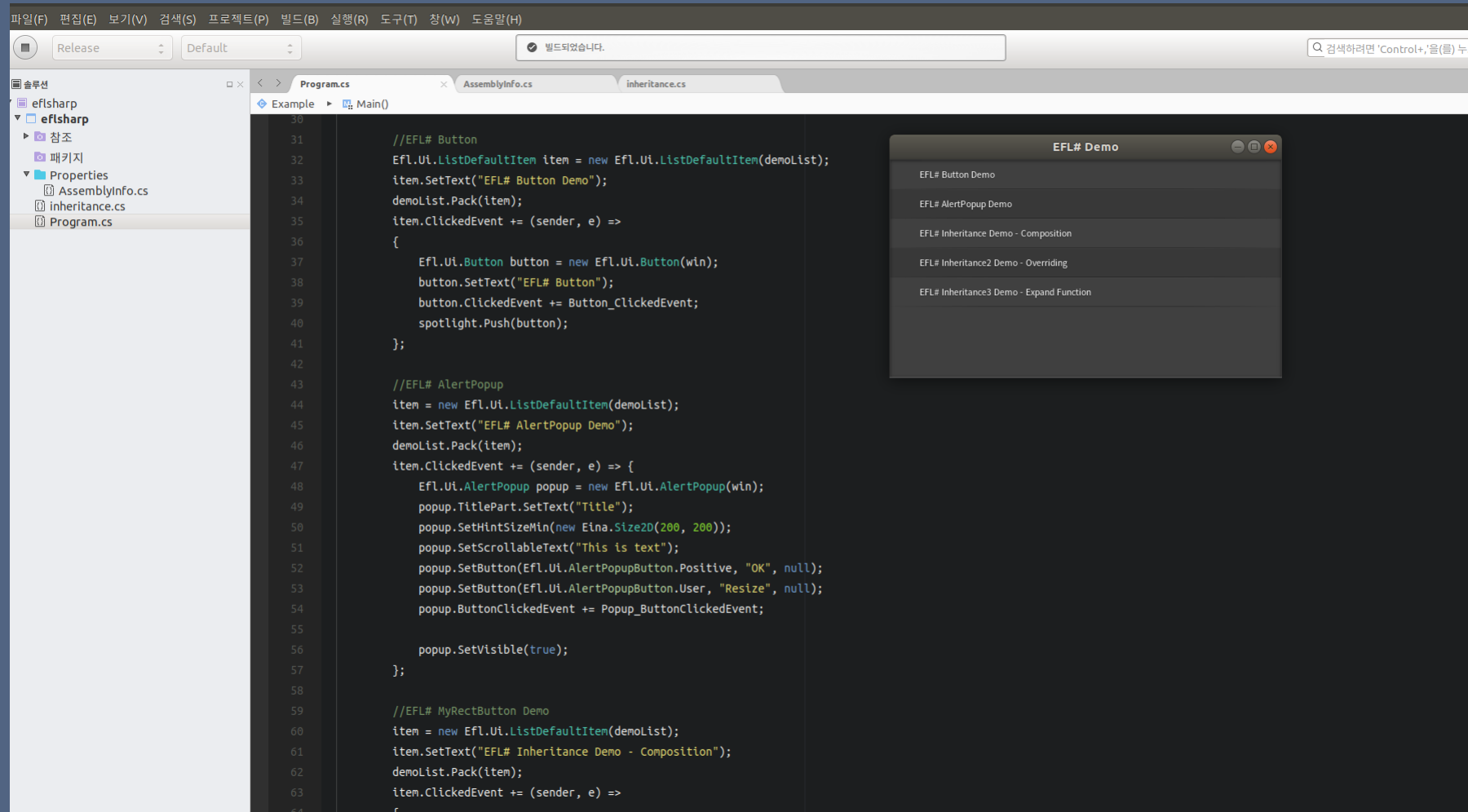
```
Eo* efl_ui_popup = efl_add(EFL_UI_ALERT_POPUP_CLASS, win);
efl_text_set(efl_part(efl_ui_popup, "title"), "title");
efl_gfx_hint_size_min_set(efl_ui_popup, EINA_SIZE2D(200, 200));
efl_ui_widget_scrollable_text_set(efl_ui_popup, "This is text");
efl_ui_alert_popup_button_set(efl_ui_popup, EFL_UI_ALERT_POPUP_BUTTON_POSITIVE, "Ok", NULL);
efl_ui_alert_popup_button_set(efl_ui_popup, EFL_UI_ALERT_POPUP_BUTTON_POSITIVE, "Cancel", NULL);
efl_event_callback_add(efl_ui_popup, EFL_UI_ALERT_POPUP_EVENT_BUTTON_CLICKED, efl_ui_text_alert_popup_clicked_cb, NULL);
```

EFL# Strengths

- Many C# developer can use EFL
- Easy to access for methods and properties (Intelligence)
- Reusability
- Expandability

Sharing Development Experience with MonoDevelop IDE

- Demo

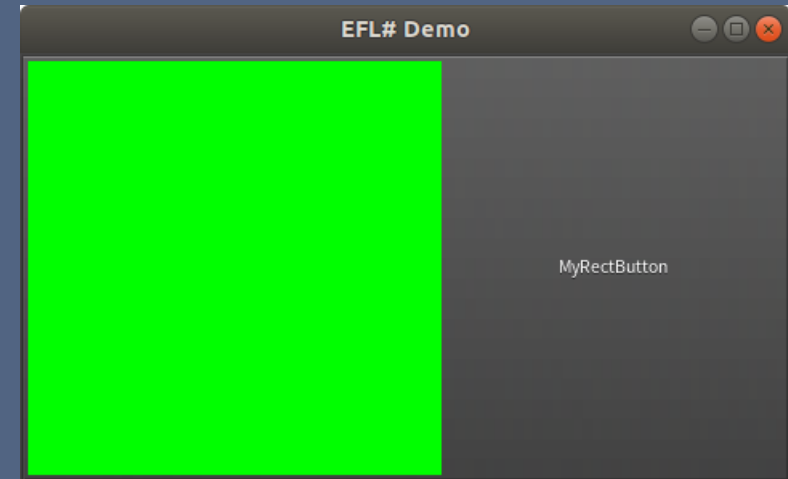


IDE

- MonoDevelop
- Visual Studio / Code
- Etc

Expand Function : MyColorRectButton

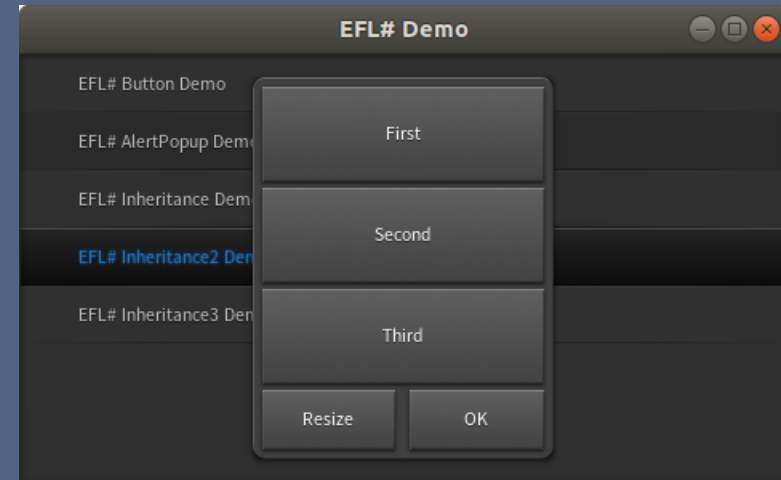
```
public class MyRectButton : Efl.Ui.Button
{
    private Efl.Canvas.Rectangle rect;
    public MyRectButton(Efl.Object parent) : base(parent)
    {
        this.SetText("MyRectButton");
        rect = new Efl.Canvas.Rectangle(this)
        {
            //Tuple
            Color = (255, 255, 255, 255)
        };
        this.SetContent(rect);
    }
    public override void SetColor(int r, int g, int b, int a)
    {
        rect.Color = (r, g, b, a);
    }
}
```



Expand Function : MyBoxAlertPopup

```
public class MyBoxPopup : Efl.Ui.AlertPopup
{
    private Efl.Ui.Box box;
    public MyBoxPopup(Efl.Object parent) : base(parent)
    {
        box = new Efl.Ui.Box(this);
        box.SetOrientation(Efl.Ui.LayoutOrientation.Vertical);
        base.SetContent(box);
    }

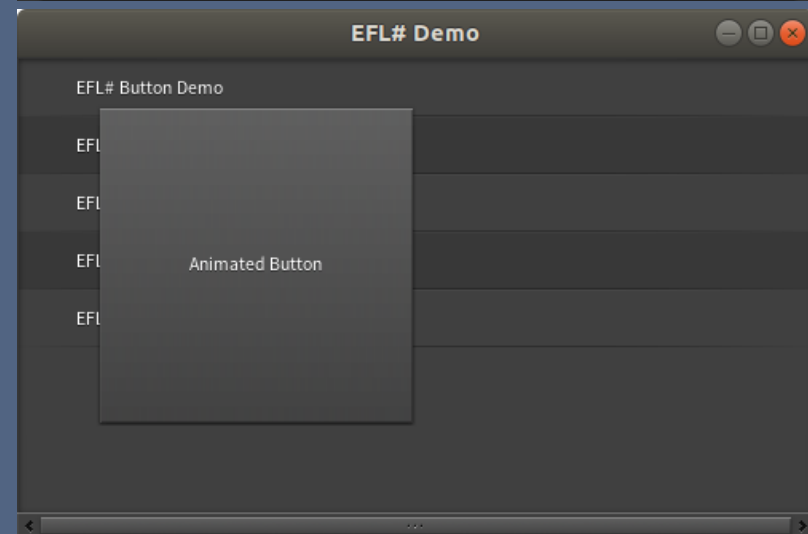
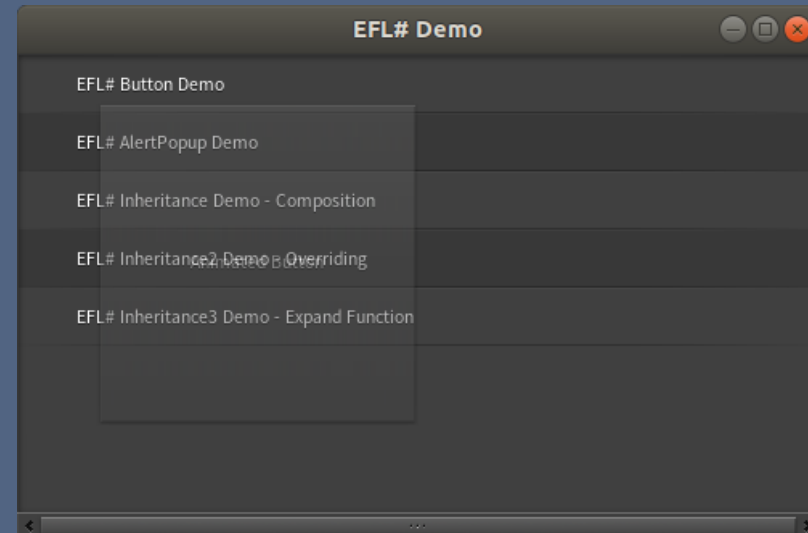
    public override bool SetContent(IEntity content)
    {
        box.PackEnd(content);
        return true;
    }
}
```



Expand Function : MyAnimatedButton

```
public class MyAnimatedButton : Efl.Ui.Button
{
    public MyAnimatedButton(Efl.Object parent) : base(parent) {}
    public override void SetVisible(bool v)
    {
        if (!Finalized) return;
        if (v)
        {
            base.SetVisible(v);

            Efl.Canvas.AnimationAlpha alpha = new Efl.Canvas.AnimationAlpha();
            alpha.SetAlpha(0.0, 1.0);
            alpha.SetDuration(0.25);
            Efl.Canvas.AnimationPlayer showPlayer = new Efl.Canvas.AnimationPlayer(this);
            showPlayer.SetAnimation(alpha);
            showPlayer.SetTarget(this);
            showPlayer.SetPlaying(true);
        }
        else
        {
            Efl.Canvas.AnimationAlpha alpha = new Efl.Canvas.AnimationAlpha();
            alpha.SetAlpha(1.0, 0.0);
            alpha.SetDuration(0.25);
            Efl.Canvas.AnimationPlayer hidePlayer = new Efl.Canvas.AnimationPlayer(this);
            hidePlayer.SetAnimation(alpha);
            hidePlayer.SetTarget(this);
            hidePlayer.SetPlaying(true);
            hidePlayer.EndedEvent += (sender, e) =>
            {
                this.Del();
            };
        }
    }
}
```



Discussion

- Layout Customization for EFL#
- Supports Proper Part Properties for EFL#
- Remove @beta tags for EFL#

