



Getting Touch Events to Play Nice in Flex

Ryan Frishberg
Software Consultant, Lab49
<http://www.frishy.com>



About Me...About You

- Former Flex SDK Engineer
- Currently consultant at Lab49 UK



Flash Touch Basics

- `flash.ui.MultitouchInputMode`
 - `TOUCH_POINT`
 - `GESTURE`
 - `NONE`



Flash Touch Basics

- `flash.events.TouchEvent`
 - BEGIN
 - END
 - MOVE
- `TouchEvent`
 - `touchPointID`
 - `isPrimaryTouchPoint`
 - `pressure`
 - `sizeX`
 - `sizeY`



Flash Touch Basics

- `flash.events.GestureEvent`
 - `GESTURE_TWO_FINGER_TAP`
- `flash.events.GestureTransformEvent`
 - `GESTURE_PAN`
 - `GESTURE_ROTATE`
 - `GESTURE_ZOOM`
 - `GESTURE_SWIPE`
- `flash.events.PressAndTapGestureEvent`
 - `PRESS_AND_TAP_GESTURE_EVENT`

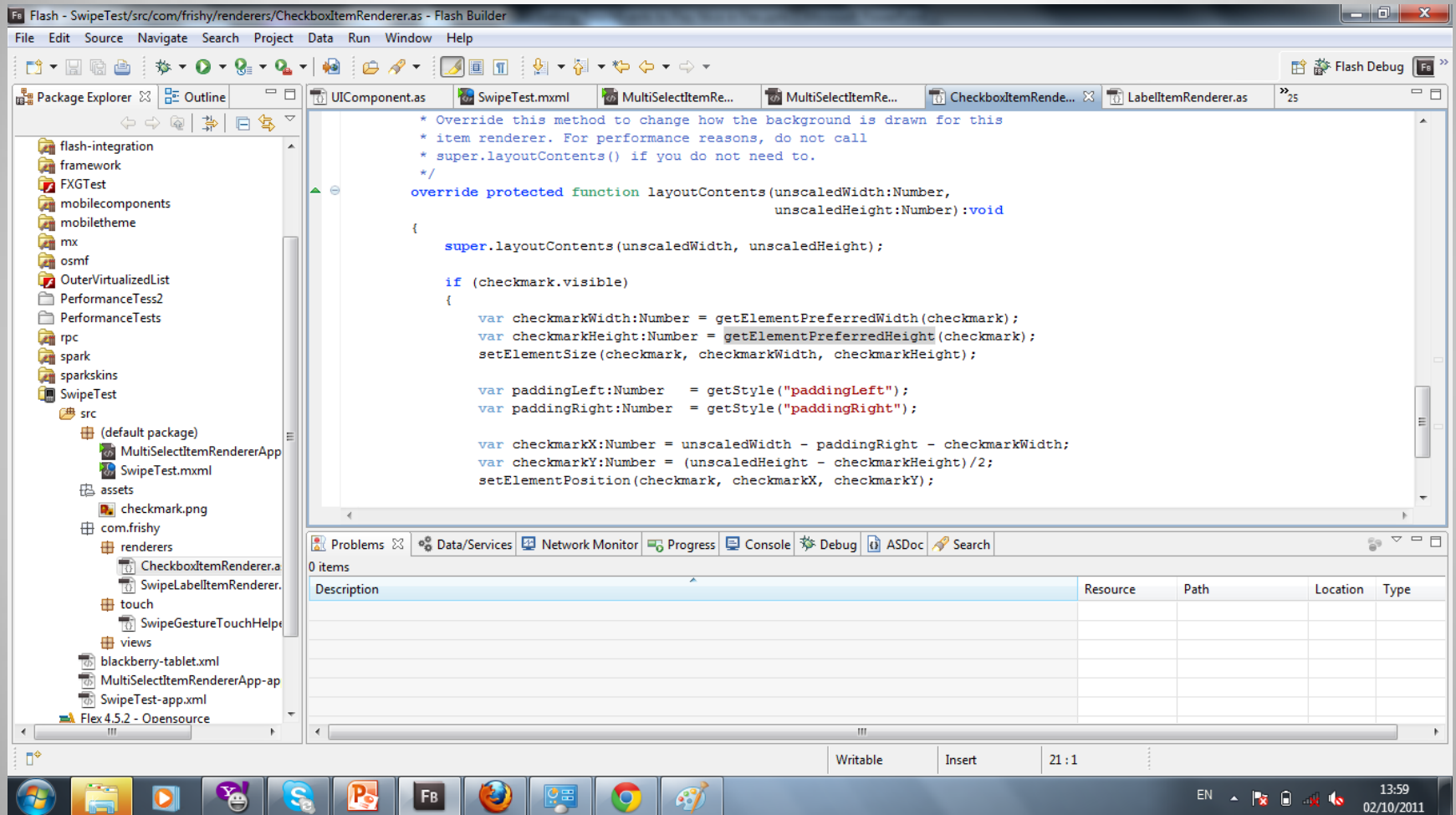


Flex Touch Basics

- `MultiTouch.inputMode` controls what Flash events you get
- `UIComponent.interactionMode` style controls how the Flex components respond to those events
 - `mx.core.InteractionMode`
 - TOUCH
 - MOUSE
- New down state for item renderers
 - Down vs selected: Down shows intent to click, selected shows item selection(s). To show both in mobile, use an item renderer with a checkmark



Multi-select Item Renderers Example



Why is Touch Difficult

- Touch events can be inaccurate
- Interactions are asynchronous and ambiguous
- Who interprets what was meant?
 - Components need to coordinate together



Button clicks in a touch-world

- User presses down on a button
- Are they attempting to touch the button or start scrolling?
- When do we show a down state on the Button?
- What if the user starts scrolling later?
- What if the user presses on the button quickly
 - how does the user receive any feedback?



Mousedown cancellation: Scrolling on a Button



1. User clicks down on the button



2. After a short delay period (around 100 ms), if the user hasn't moved his finger enough to start a scroll, then the button shows up as visibly mouseDowned.



3. Once the user moves enough pixels to register a scroll, the button returns to its normal state and a scroll starts. The amount of pixels needed to move to register a scroll is called the “slop.”

Mousedown cancellation: Scrolling on a Slider



Touch-cancellation Behaviors

- MouseEvents are bubbled up.
- When a component sees a mouse event or series of mouse events they want to interpret, the component should dispatch TouchInteractionEvents
- Three TouchInteractionEvents:
 - STARTING, START, END



Scrolling Notification Implementation: Part 1

- User clicks on button
- Button receives the mouseDown event. The Button starts a Timer and waits for the touchDelay (by default, 100 ms) before going in to the mouseDown state.
- Scroller receives the bubbled up mouseDown event and attaches mouseMove and mouseUp listeners to watch for a vertical scroll user gesture.



Initial Button state after mousedown



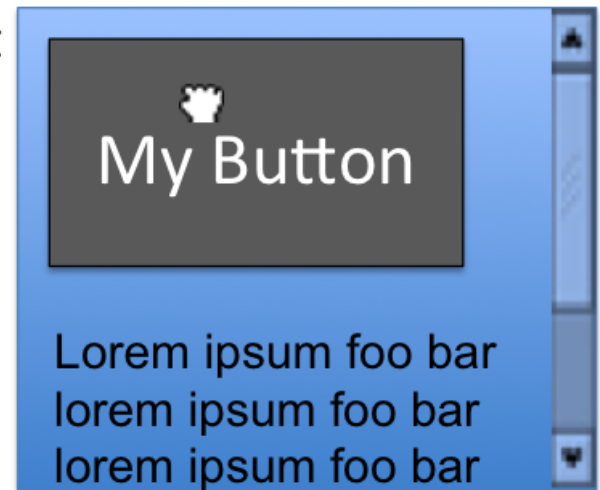
Button state after small delay

Scrolling Notification Implementation: Part 2

- User moves around enough for Scroller to consider the movement the start of a scroll operation
- Scroller dispatches a bubbling, cancellable `touchInteractionStarting` event to the originally mousedowned display object (the button in this case)
- The button sees the `touchInteractionStarting` event and doesn't do anything. (if it wanted to keep control of this user gesture, it would call `event.preventDefault()`, like Slider might in some cases)
- The Scroller dispatches a bubbling `touchInteractionStart` event to the originally mousedowned display object. The Button sees this event and gives up control of this user-gesture by getting out of the mousedowned visual state.



Button state after small delay



Button State after scroll has started

Scrolling Notification Implementation: Part 3

- Now that the Scroller has started scrolling, the Scroller handles the vertical scroll operation.
- When the user mouses up, the Scroller will dispatch a `touchInteractionEnd` event immediately if the user hasn't thrown the list. However, if the user moved quickly enough to generate a throw, the `touchInteractionEnd` event will be dispatched at the end of the throw.



Button State during scroll

Buttons inside item renderers

- Not just a touch issue! Issues occur with mouse as well
- `event.preventDefault()` is your friend



Adding Swipe Gesture

- Listen for GestureEvent
- Listen for other TouchInteraction STARTs and ENDs – can't interrupt another START
- Once swipe occurs, dispatch a STARTING, and if not cancelled, dispatch a START (and then an END since it's not asynchronous)



Questions

