

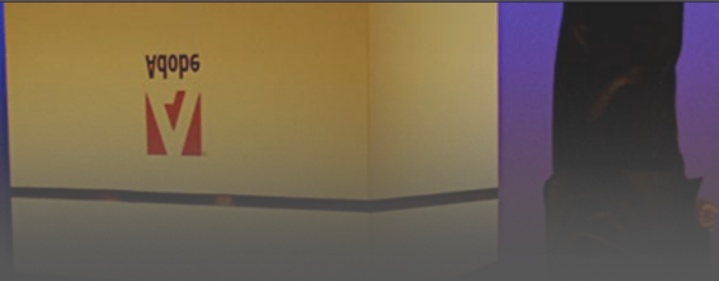
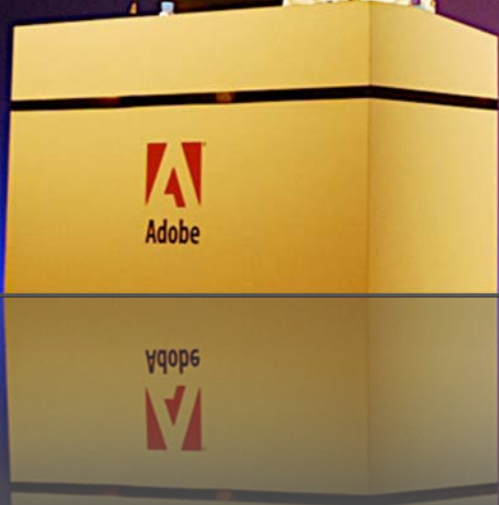
# 詳説アイテムレンダラー

Flex User Group

廣畑 大雅 (taiga.jp)

有川 榮一 (AKABANA)

```
1 import flash.events.*;
2 import flash.events.MouseEvent;
3 import mx.core.UIComponent;
4
5 public class DrawCanvas extends UIComponent
6 {
7     private var _color:uint;
8
9     public function get color():uint{
10         return _color;
11     }
12
13     public function set color(value:uint):void{
14         if( _color !=
```



# アジェンダ

- 主要コンポーネント
- 実装方法
- 内部動作
- カスタムアイテムレンダラー
- アイテムレンダラー機能を独自実装

# 主要コンポーネント

- Accordion
- ComboBox
- DataGrid
- List
- Menu
- Tree

# 主要コンポーネント

- Accordion      ->      AccordionHeader
- ComboBox      ->      ListItemRenderer
- DataGrid      ->      DataGridItemRenderer
- List      ->      ListItemRenderer
- Menu      ->      MenuItemRenderer
- Tree      ->      TreeItemRenderer

# 実装方法

- タグで定義

```
<mx:DataGridColumn...>  
  <mx:itemRenderer>  
    <mx:Component>  
      <mx:HBox>  
        <mx:TextInput />  
        <mx:Button />  
      </mx:HBox>  
    </mx:Component>  
  </mx:itemRenderer>  
</mx:DataGridColumn>
```



deprecated

理由

- ・ドキュメント作成しづらい
- ・ソース管理しづらい
- ・汎用的でない

# 実装方法

- タグの itemRenderer プロパティに定義 1

```
– <mx:DataGridColumn  
  ...  
  itemRenderer = “jp.taiga.renderer.MyRenderer”  
  ...  
/>
```

- タグの itemRenderer プロパティに定義 2

```
– <mx:DataGridColumn  
  ...  
  itemRenderer = “{new ClassFactory(MyRenderer)}”  
  ...  
/>
```

# 内部動作

- itemRenderer プロパティの引数の型は IFactory
  - 例 : Accordion.as

```
public function Accordion() {  
    super();
```

```
    headerRenderer  
        = new ClassFactory(AccordionHeader);
```

```
...
```

```
public function set headerRenderer(value:IFactory):void {  
    _headerRenderer = value;
```



# 内部動作

- itemRenderer プロパティの引数の型は IFactory
  - 例 : List.as

```
public function set itemRenderer(value:IFactory):void {  
    _itemRenderer = value;  
  
    invalidateSize();  
    invalidateDisplayList();  
  
    itemsSizeChanged = true;  
    rendererChanged = true;  
  
    dispatchEvent(new Event("itemRendererChanged"));  
}
```

# 内部動作

```
package mx.core {
    public class ClassFactory implements IFactory {
        public var generator :Class;
        public var properties :Object = null;
        public function ClassFactory(generator:Class = null) {
            super();
            this.generator = generator;
        }
        public function newInstance():* {
            var instance:Object = new generator();
            if (properties != null) {
                for (var p:String in properties) {
                    instance[p] = properties[p];
                }
            }
            return instance;
        }
    }
}
```

# 内部動作

- インスタンス生成
  - 例 : List.as

```
override public function  
createItemRenderer(data:Object):IListItemRenderer {  
  
    var factory:IFactory;  
  
    ...  
  
    if (!renderer) {  
  
        renderer = factory.newInstance();  
  
        ...  
    }  
}
```

# 内部動作

- インスタンス表示
  - 例 : List.as

```
protected var listContent:ListBaseContentHolder;
...
override protected function
makeRowsAndColumns(···):Point {

...
if (!item) {
    item = createItemRenderer(data);
    item.owner = this;
    item.styleName = listContent;
    listContent.addChild(DisplayObject(item));
}
```

# カスタムアイテムレンダラー

- インターフェース
  - mx.core
    - IDataRenderer
  - mx.controls.listClasses
    - IDropInListItemRenderer
    - IListItemRenderer
  - mx.managers
    - IFocusManagerComponent

# カスタムアイテムレンダラー

- カスタム DataGridView アイテムレンダラーサンプル
- カスタム List アイテムレンダラーサンプル

# 内部動作

- itemRenderer 発行のイベント
  - イベントクラス ( またはサブクラス ) の bubbles を true に

```
public class MyCunstomEvent extends flash.events.Event
{
    ...
    public function MyCunstomEvent(
        type      :String,
        bubbles   :Boolean = true,
        cancelable :Boolean = false )
    {
        super(type, bubbles, cancelable);
    }
    ...
}
```

# アイテムレンダラー機能を実装

- 例 : itemRenderer プロパティを実装したサンプル