JavaOne℠

MobiTV: Creating Effective Mobile Content Now and in the Future

Roy Ben Hayun
Sun Microsystems
System Architect

Do Hyun Chung
MobiTV
Lead Software Engineer

We will go on a tour...

Take a few stops...

Talk about what we see there...

And, watch TV...

Agenda

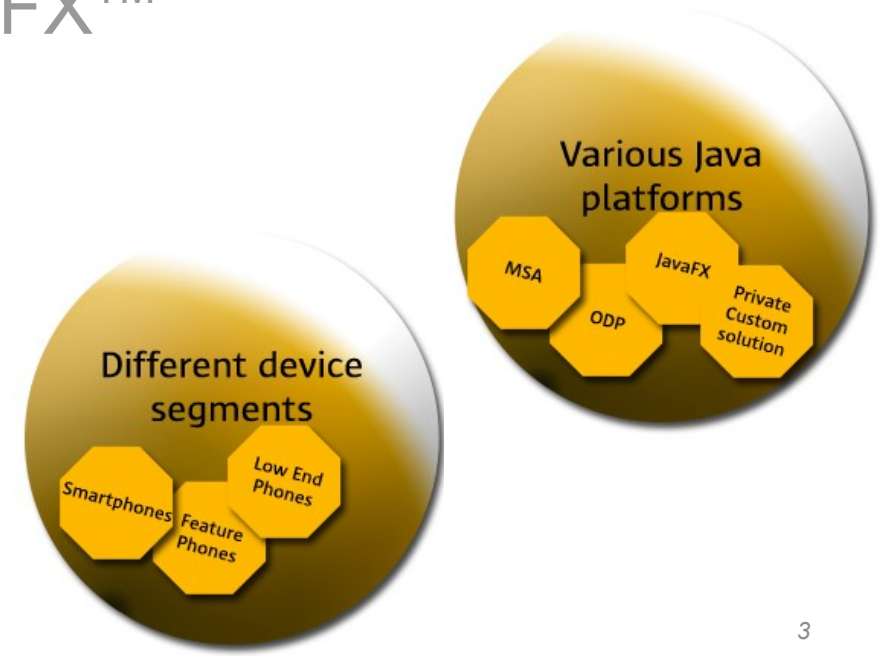
Overview on Java™ and Mobile TV

Targeting mass market with JTWI / MSA

Targeting carriers with ODP

Targeting high-end with JavaFX™

Pre-integration with SJWC



Mobile TV

Mobile TV Reach

- 59m Broadcast Mobile TV handsets in 08
- MobiTV announced 6m subscribers in April 09 and streamed in excess of 1billion minutes in 08
- Expansion of 3G/4G networks and flat rate data plans will combine to drive mobile TV subscribers to 462 million worldwide by 2012 (ABI REsearch)

Technology Trends

- Increased smartphone adoption
- Hybrid delivery
- Broadcast and Unicast services
- Personalization is key
- Interactive TV

Mobile TV Needs

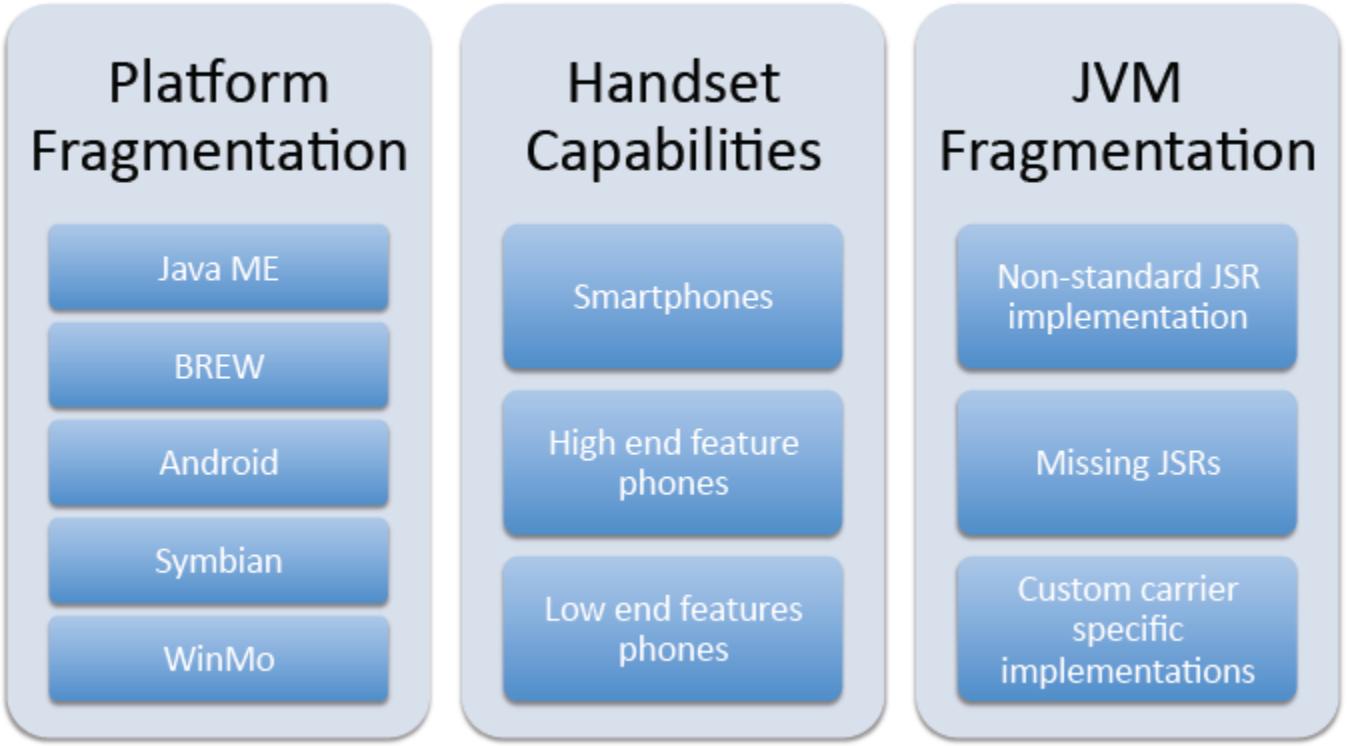
Better user experience

- Fast channel change
- Video Quality
- Improved content discovery
- Personalization & customization

Monetization of service

Hybrid Mobile TV Ecosystem

Mobile Handset Challenges



Agenda

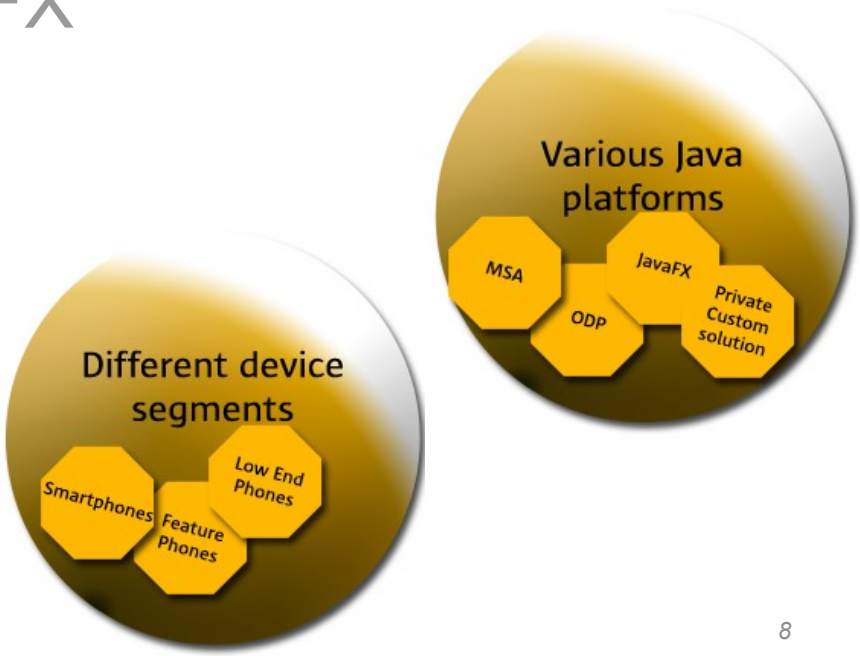
Overview on Java and Mobile TV

Targeting mass market with JTWI / MSA

Targeting carriers with ODP

Targeting high-end with JavaFX

Pre-integration with SJWC



Overview of streaming media

How is the content packaged?

Containers – what are they really?

e.g., 3gp, mov, mpg

Video codecs

e.g., RealVideo, H.263, H.264, MPEG-4

Audio codecs

e.g., RealAudio, AMR, AAC

Overview of streaming media

How is the content delivered?

RTSP – Real Time Streaming Protocol

- Control protocol e.g., PLAY, STOP, PAUSE

RTP – Real-Time Transport Protocol

- The media payload over TCP or UDP

HTTP

- TCP transport
- Entire clip is buffered!
- Progressive downloading – Not widely implemented

Multimedia challenges in Java
Under the hood

Java acts as a bridge to the OEM codec support

- Dependencies across multimedia layers on device

No fine-grained control on the data

- Cannot do a fast change of TV channel (aka SFCC)

How to identify the failing point?

- Testing media resources across the multimedia layers
- Network sniffers (e.g., Ethereal)

Basic video playback with JSR-135

And remember – should run in its own Java thread!

```
Manager.createPlayer("rtsp://.../movie.3gp");
dPlayerListener(this); //TODO: handle events
alize();
 (VideoControl) p.getControl("VideoControl");
 append((Item)vc.
  initDisplayMode(vc.USE_GUI_PRIMITIVE, null));
efetch();
art();
```

Utilizing JSR-135 detection

Detecting and utilizing

```
//e.g., "rtsp" (or null)
String[] supportedContentTypes =
  Manager.getSupportedContentTypes(protocol);


//e.g., "audio/x-wav" (or null)
String[] supportedProtocols =
  Manager.getSupportedProtocols(contentType);


//e.g., "video.encodings", "streamable.contents"
String val = System.getProperty(mmapiKey);
```

MobiTV's MIDP application

Show me some real code :-)

Use JSR-135 for media playback

Xml based channel lineup info from Server

Application Framework Library

- Canvas based UI
- Code reuse
- Hides device differences

MobiTV MIDP application - MicroCodes

One idea to cope with device fragmentation.

```
protected static final char CONSTRUCT        = 'n';
protected static final char REALIZE          = 'r';
protected static final char PREFETCH         = 'p';
protected static final char SET_VISIBLE      = 'v';
protected static final char START            = 's';
public static final String MICROCODE_DEFAULT ="nrwxdcpizlsv";

// redefine microcode for each device to cope with the differences
int processMicrocodes( String codes, Item item ) {
for( int i = 0; i < codes.length(); i++ ) {
switch( code ) {
case CONSTRUCT:
// create player
...
```

Demo – MobiTV's MobileTV MIDP app

Using LWUIT MediaComponent

```
player =
    Manager.createPlayer("rtsp://../movie.3gp");
player.setLoopCount(-1);
player.realize();
//wrapper around JSR-135 Player
media = new MediaComponent(player);
media.setWidth(...);
media.setHeight(...);
//add to container with center constraint
form.addComponent(BorderLayout.CENTER, media);

media.start();
```

Agenda

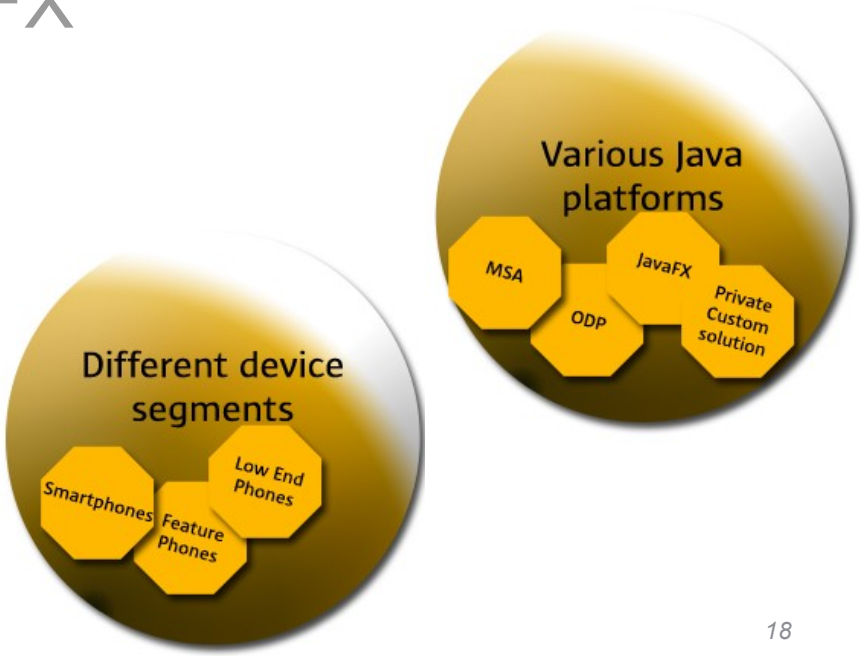
Overview on Java and Mobile TV

Targeting mass market with JTWI / MSA

Targeting carriers with ODP

Targeting high-end with JavaFX

Pre-integration with SJWC



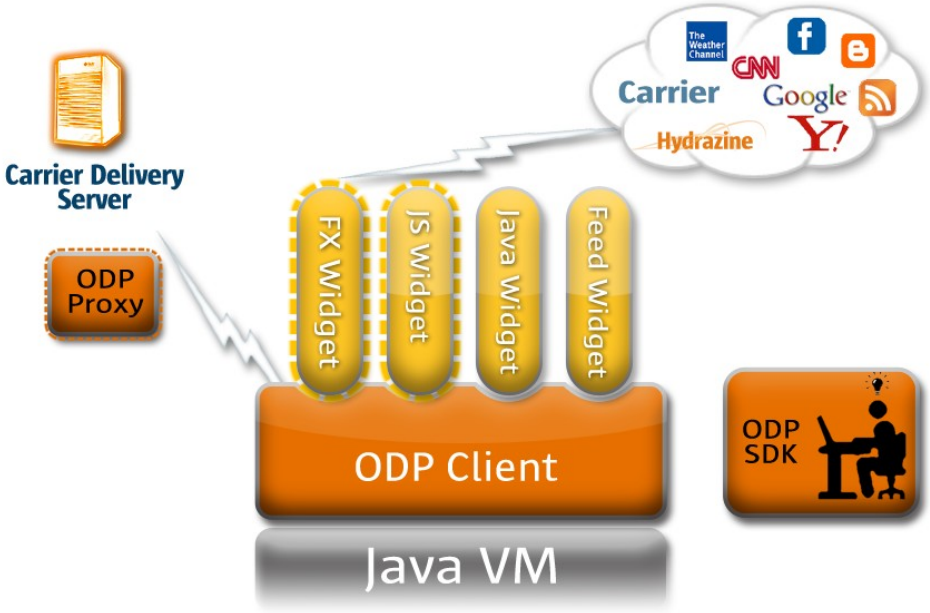
Targeting carriers with On Device Portal

Java Widgets (based on LWUIT)

The home screen

The idle screen

Shopping client



Turning LWUIT MIDlet into an ODP widget

Three simple steps (see ODP documentation)

Subclassing the WIDletBase

```
public void start(WIDletContext context);
public void stop();
public Component getPreviewScreen();
public String getPreviewTitle();
public void goToBackground\Foreground();
...
```

MobiTV's ODP widget

creating the mediaForm

```
ublic void showMediaForm(int index) {
 mediaForm = new WIDletForm(context) {
     //handle "Back" action; show previous form
     public void showPreviousForm() {
             if(player != null && media != null &&
                     player.getState() != player.CLOSED) {
                 //stop playback
                 media.stop();
                 player.close();
             }
             //call override method
             super.showPreviousForm();
     }
 };
 mediaForm.setLayout(new BorderLayout());
```

Demo – ODP and MobiTV's Widget

Agenda

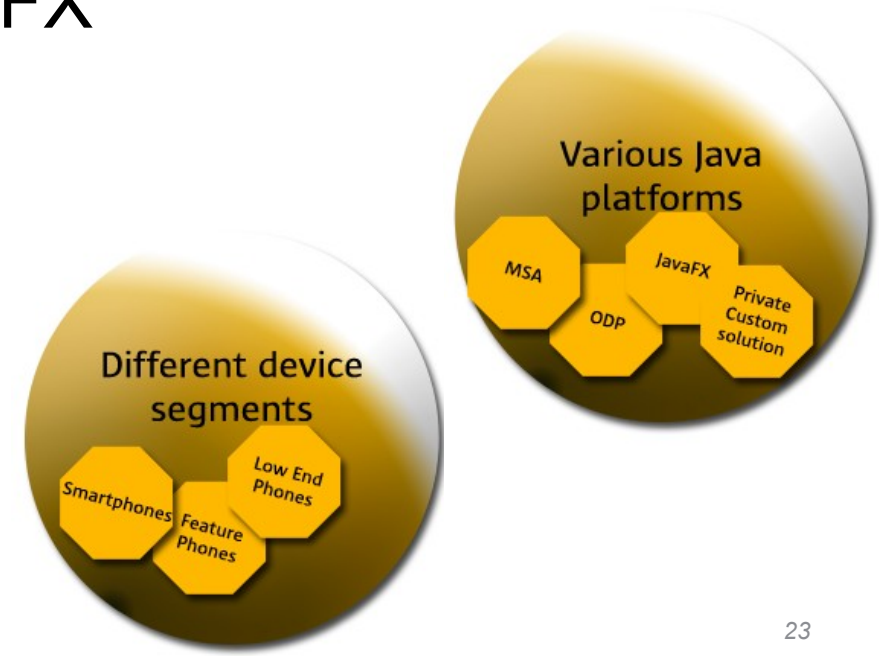
Overview on Java and Mobile TV

Targeting mass market with JTWI / MSA

Targeting carriers with ODP

Targeting high-end with JavaFX

Pre-integration with SJWC



Targeting high-end with JavaFX

RIA fancy transitions and special effects

Java FX script

- Scene graph

Easily mix Java and FX

- Java ME backend
- FX frontend

Hello, FX World!

```
var logo:Text = Text{
                font : Font { size : 24 }
                fill: Color.RED
                content: "Hello, FX World!" };

Stage { title: "FX MobileTV app"
        width: 250, height: 320
        scene: Scene {
        content: [ screenNode, logo ] }
    }
```

Using FX Media package

```
yer = MediaPlayer { //FX MediaPlayer
ia: Media { source: bind src }; //FX Media
oPlay:false }
diaView
iaView = MediaView{ mediaPlayer:player }


yLocal = ImageButton {
n: function():Void {
rc = "rtsp:// . /clip.3gp";//set Media.source
layer.play(); } }; //play clip
```

```
public class ScreenNode extends CustomNode{
  //FX Media, Player and View
  ...
```

Adding the Media UI to the scene graph

```
  override public function create():Node {
    var g = Group {
        content: [ mediaView,
                   playLocal, stopLocal,]};
      return g;}
}
```

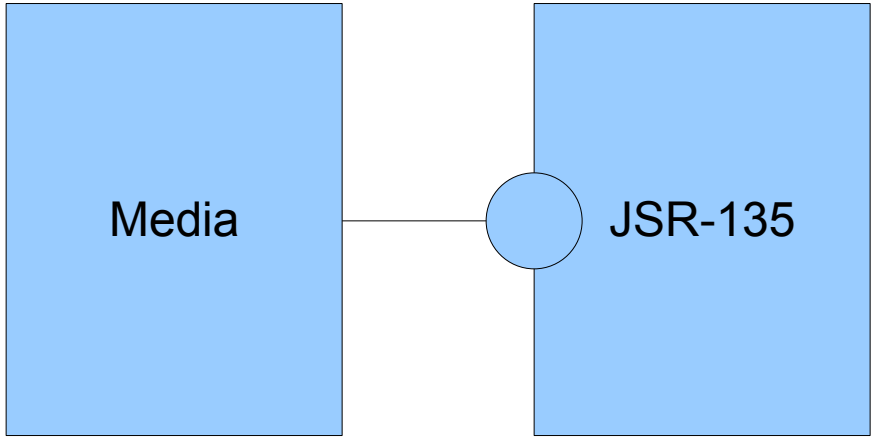
Under the hood of FX Media package

FX is implemented on top of Java ME

An FX media object has a Java Peer

FX Media delegates to Java JSR-135

Instantiation of JSR-135 Players



Fancy screen transition on FX

```
def blindRight = TranslateTransition {
    node: bind Main.deckRef.visibleNodeRef
    duration: 0.2s
    fromX: -240
    toX: 0
    repeatCount: 1
    autoReverse: true
}
// keep the transition object as a member
node.transition = blindRight;
// later, do the transition as needed over and over
node.transition.playFromStart();
```

Code Sample: Mix up FX and Java

```
//in MobiUtilsProxy.java
public class MobiUtilsProxy {
    static MobiUtils mobi = MobiUtils.getInstance();
    static final boolean dummy = true;
    public static void initialize() {
        if(dummy == false ) mobi.mobiInit();
    }
    public static void play(String url) {
        if(dummy == false ) mobi.play(url);
    }
}
//in Playback.Screen.fx
MobiUtilsProxy.initialize();
MobiUtilsProxy.play("rtsp://some.server/movie.3gp");
```

Demo – MobiTV's FX app

Agenda

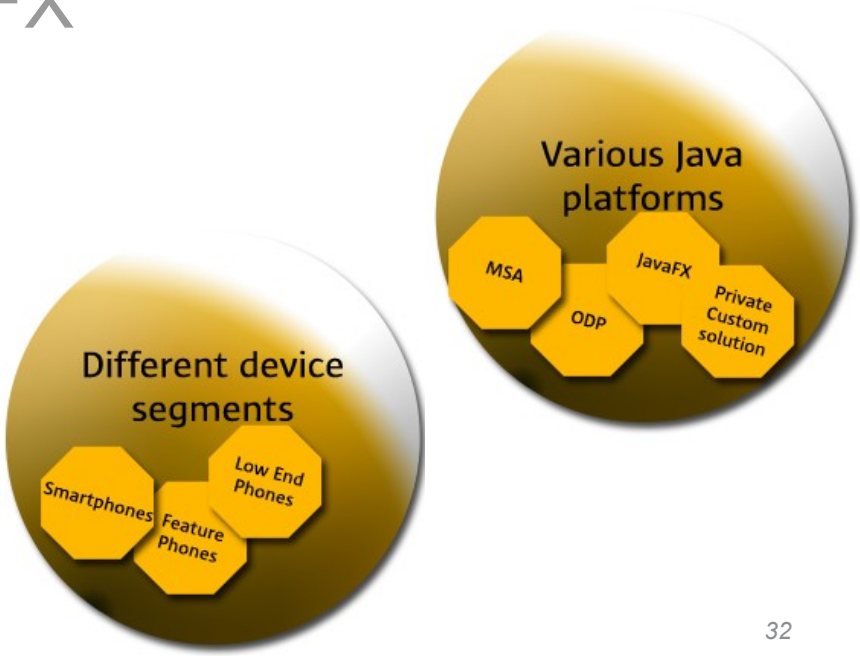
Overview on Java and Mobile TV

Targeting mass market with JTWI / MSA

Targeting carriers with ODP

Targeting high-end with JavaFX

Pre-integration with SJWC



MobiTV's Super Fast Channel Change

SFCC is a killer feature by MobiTV

- Immediate change of TV channel – Minimal delays
- Enhances UX tremendously

Challenges in Java ME

- No fine-grained control over the RTSP\RTP streams

MobiTV's native WM solution

- Proprietary RTSP module - WM DSHOW Filter

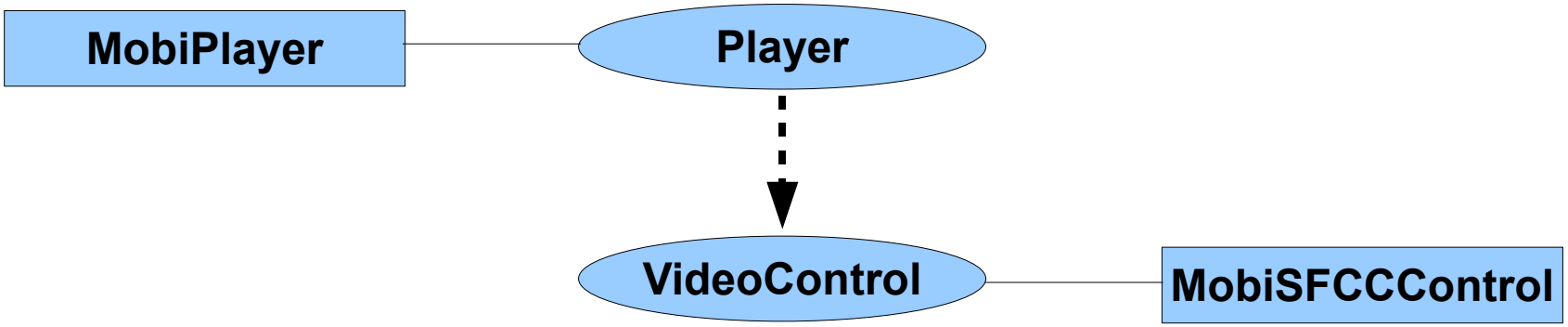
SFCC on JavaFX Mobile

Advanced use-case by collaboration with Sun

- Implementation options
  - Fully embedded RTSP/RTP stack in the SJWC
  - Integrating MobiTV's developed solution
    - JSR-135 oriented approach
    - Private Custom APIs for MobiTV
- Challenges
  - Exposing all required capabilities for MobiTV's service
  - Exposing SFCC on MIDP, ODP and JavaFX

JSR-135 oriented approach

Custom VideoControl, Player and URI scheme



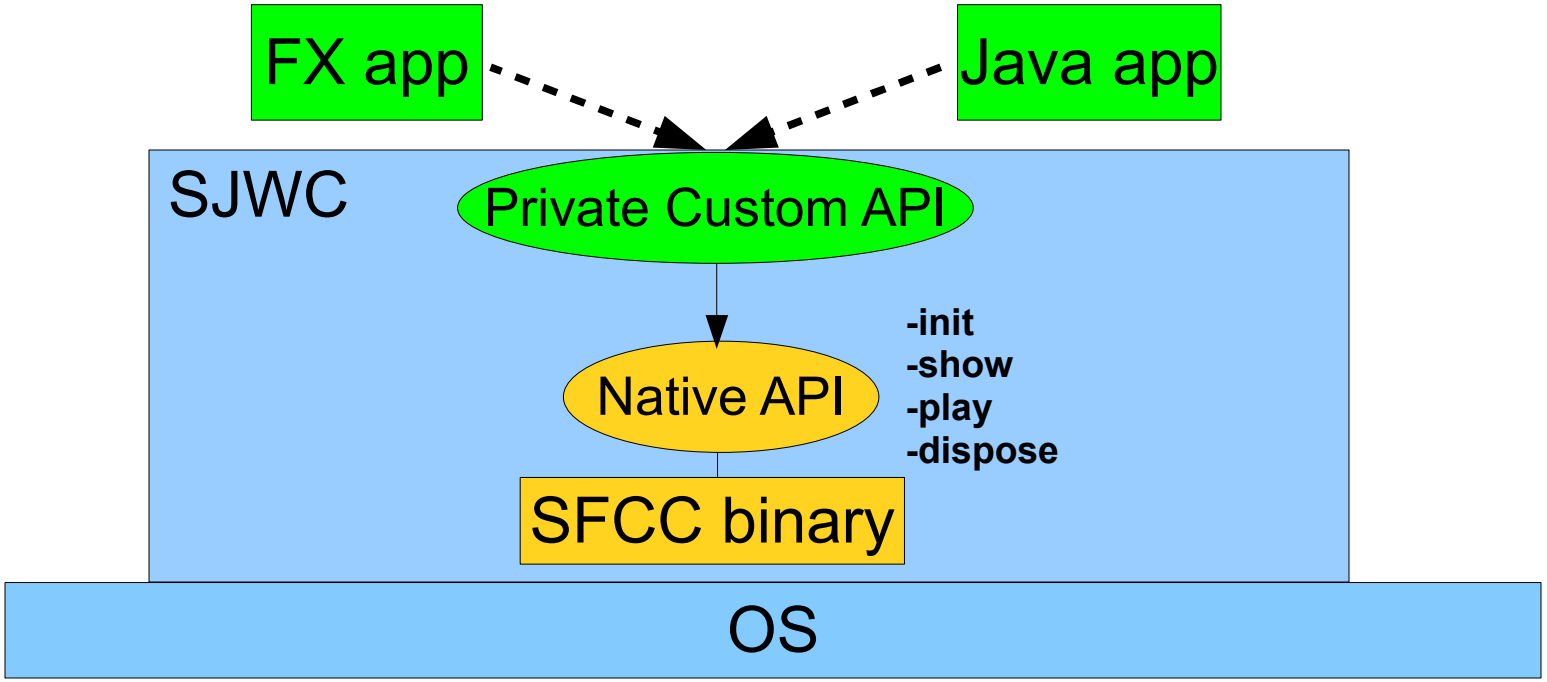

```
p = Manager.createPlayer("mobirtsp:// ... ");

p.start();

...

mvc = (MobiSFCCControl)
  p.getControl("com.mobitv.sfcc.MobiSFCCControl");

mvc.switchChannel("mobirtsp:// ...");
```

Private Custom design approach

Private Custom APIs to native SFCC support

Works the same for MIDP and for FX

JSR-135 still used for local playback



Private Custom Java API for MobiTV

```
Public class MobiApiUtils {
    //initialize native SSFC module
    public void mobiInit();
    //play or switch channel with SSFC support
    public void switchChannel("rtsp:// .... ");
    //show or hide video display area
    public void show(bool);
    //move or resize video display area
    public void move(x, y, width, height);
```

Calling Private Custom API, from JavaFX

```
r nextChannel_SFCC = ImageButton {
iconUrl: "nextChannel.png"

onMouseReleased: function( e: MouseEvent ):Void {
  //use Private Custom Java API
  MobiApiUtils.getInstance().show(true);
  MobiApiUtils.getInstance().switchChannel(next);
};
```

Demo – JavaFX and MobiTV's SFCC

Summary - Connecting the dots

Java as a strategic platform for mobile TV

Covering a mobile service for all device ranges

- **Low end: JTWI (+ LWUIT)**
- **Mid -> high end : MSA (+ LWUIT)**
- **high-end -> smartphones: FX**

Using various Java technologies

- **MIDP or FX application**
- **Java ODP Widget**
- **Private Custom API by collaboration with Sun**

More information

www.mobitv.com

www.en.wikipedia.org/wiki/Mobile_TV

www.javafx.com

www.java.sun.com/products/midp

www.java.sun.com/products/java-odp

www.jcp.org

www.lwuit.dev.java.net

Roy Ben Hayun
roy.ben.hayun@sun.com

Do Hyun Chung
dchung@mobitv.com