



Java is a trademark of Sun Microsystems, Inc.

MX

DIGITAL ENTERTAINMENT TECHNOLOGY

JavaOneSM

Creating Games for Blu-ray Disc in BD-Java™

Bill Foote

Sun Microsystems

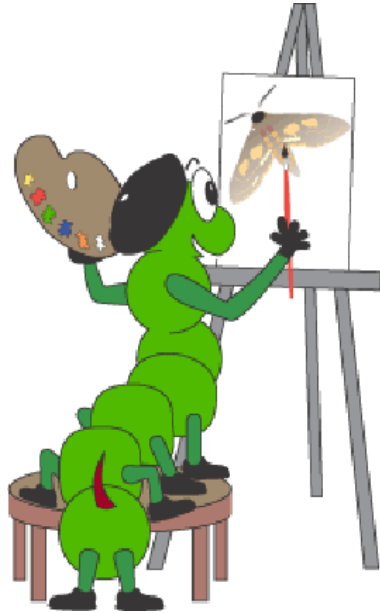
Philip C. Starner

MX Productions

Talk Outline

- > Production Roles
- > Blu-ray Basics
- > GRIN Framework
- > Game Basics
- > Example: Run and Jump Puzzle Game
- > Performance Profiling

BD-J Production Roles



Graphics Designer
Interaction Designer

Works with visual media
Creates assets, interaction flows
Might create scene graph



Disc Author
Scripting Programmer

Authors disc
Manages resource budgets
Uses scene graph intensively
Writes scripting code in Java
(single-threaded)



BD-J Non-production Role



Software Engineer/Architect

- Creates tools and frameworks
- Builds workflow system
- Deals with multi-threading
- Makes network/server architecture

Blu-ray Basics

Making a project with HD Cookbook

- > “GrinXlet” framework makes single-xlet disc
 - Make a directory with sub-directories “src”, “xlet_src” and “se_src”
 - Copy `build.xml` and `vars.properties` from `<cookbook>/xlets/GrinXlet`
 - Set a handful of properties in `vars.properties`
 - If needed, create `user.vars.properties` to point to your stubs and cookbook repository.
 - Don't check in `user.vars.properties`
 - Write your code in `src`, and possibly `xlet_src` and `se_src`
 - Type “ant”
 - Creates BD disc image + JavaSE “grinview” version

Blu-ray Basics

Making a project with HD Cookbook

```
billf@~/java.net/hdcookbook/xlets/demos/twitterGRIN$ ll
total 32
8 -rw-r--r--  1 billf  501  2988 Jan 29 17:06 LICENSE.txt
8 -rw-r--r--  1 billf  501   409 Mar 14 19:42 README.txt
8 -rw-r--r--  1 billf  501  1894 Jan 28 15:54 build.xml
0 drwxr-xr-x  3 billf  501   102 Jan 30 11:23 se_src
0 drwxr-xr-x  9 billf  501   306 Mar 16 10:33 src
8 -rw-r--r--  1 billf  501  1420 Jan 29 17:27 vars.properties
0 drwxr-xr-x  4 billf  501   136 Apr 27 12:48 xlet_src
billf@~/java.net/hdcookbook/xlets/demos/twitterGRIN$ ll src
total 72
16 -rw-r--r--  1 billf  wheel  5372 Mar 16 10:31 NetworkManager.java
24 -rw-r--r--  1 billf  501    9232 Mar 16 10:32 TwitterDirector.java
16 -rw-r--r--  1 billf  501   5944 Mar 16 10:33 TwitterPoll.java
 0 drwxr-xr-x  5 billf  501    170 Jan 30 11:23 com
 0 drwxr-xr-x  9 billf  501    306 Jan 30 11:23 images
16 -rw-r--r--  1 billf  501   6785 Jan 29 17:09 twitter_show.txt
billf@~/java.net/hdcookbook/xlets/demos/twitterGRIN$ ll se_src
billf@~/java.net/hdcookbook/xlets/demos/twitterGRIN$ ll xlet_src
total 16
16 -rw-r--r--  1 billf  501  5743 Apr 27 12:48 TwitterXlet.java
```

Blu-ray Basics

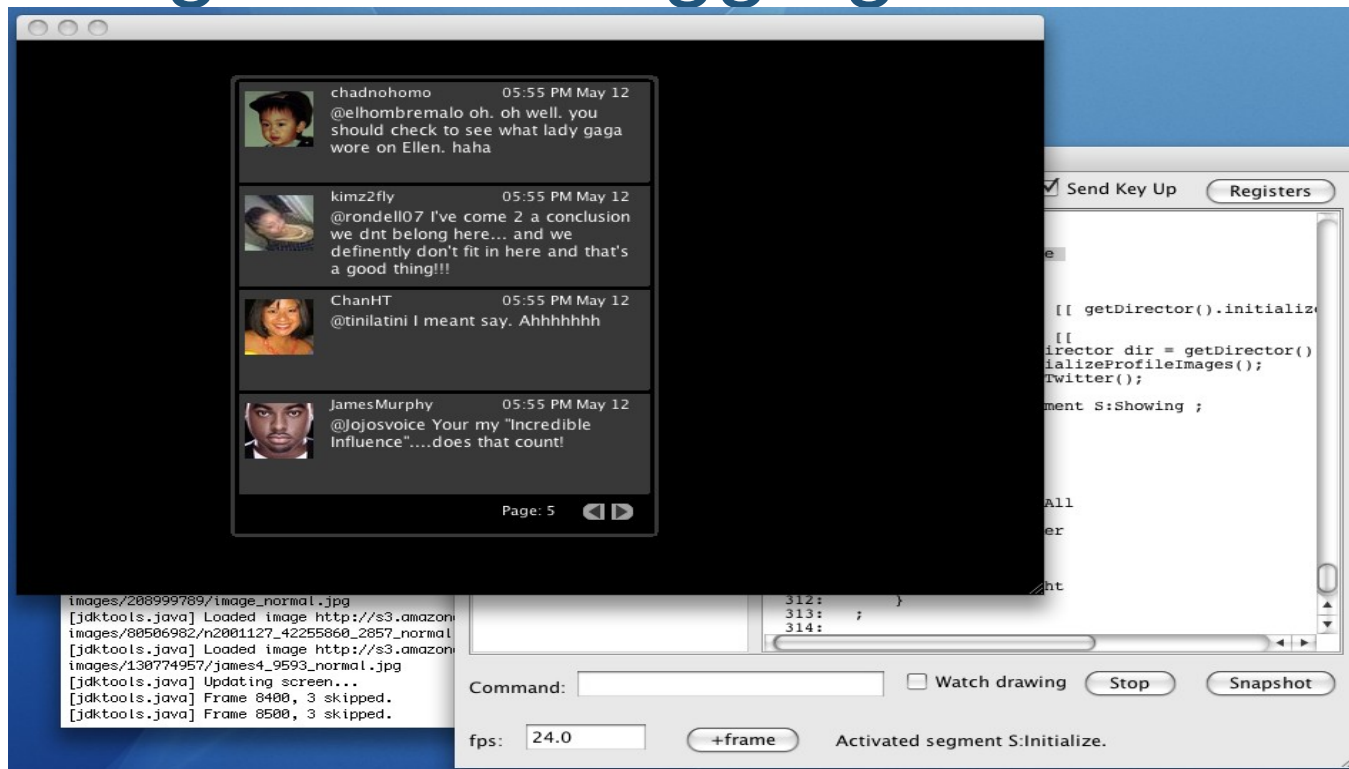
Making a project with HD Cookbook

- > GrinXlet's disc image is built with:
 - An ant build script that makes the disc directory structure
 - javac
 - GRIN scene graph compiler (converter.jar)
 - BD-J JAR signer (security.jar + bouncycastle.jar)
 - BD certificate generator (security.jar + bouncycastle.jar)
 - BDJO generator (bdjo.jar)
 - BDMV ID file generator (id.jar)
 - BDMV index file generator (index.jar)
 - BDMV MovieObject.bdmv generator (movieobject.jar)

- > A more advanced project requires setting up the build “manually”
 - E.g. for a multiple xlet disc

Blu-ray Basics

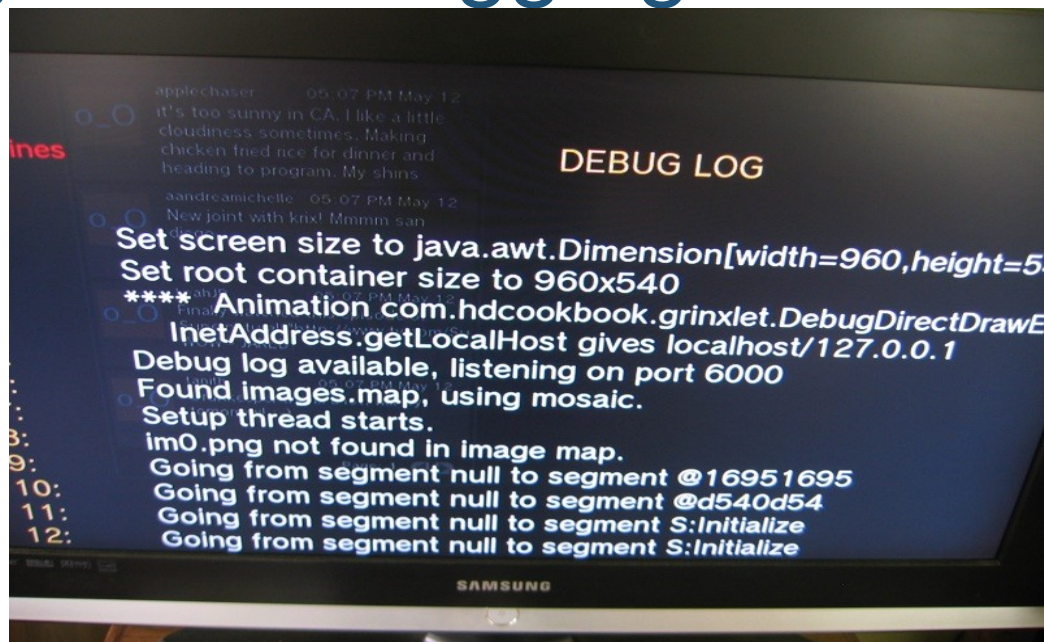
Testing and Debugging on a PC



billf@~/java.net/hdcookbook/xlets/demos/twitterGRIN\$ ant grinview
Runs xlet in a JavaSE emulation environment

Blu-ray Basics

Testing and Debugging on a Player



```
if (Debug.LEVEL > 1) {  
    Debug.println("**** Animation " + this  
        + " state set to " + s);  
}
```

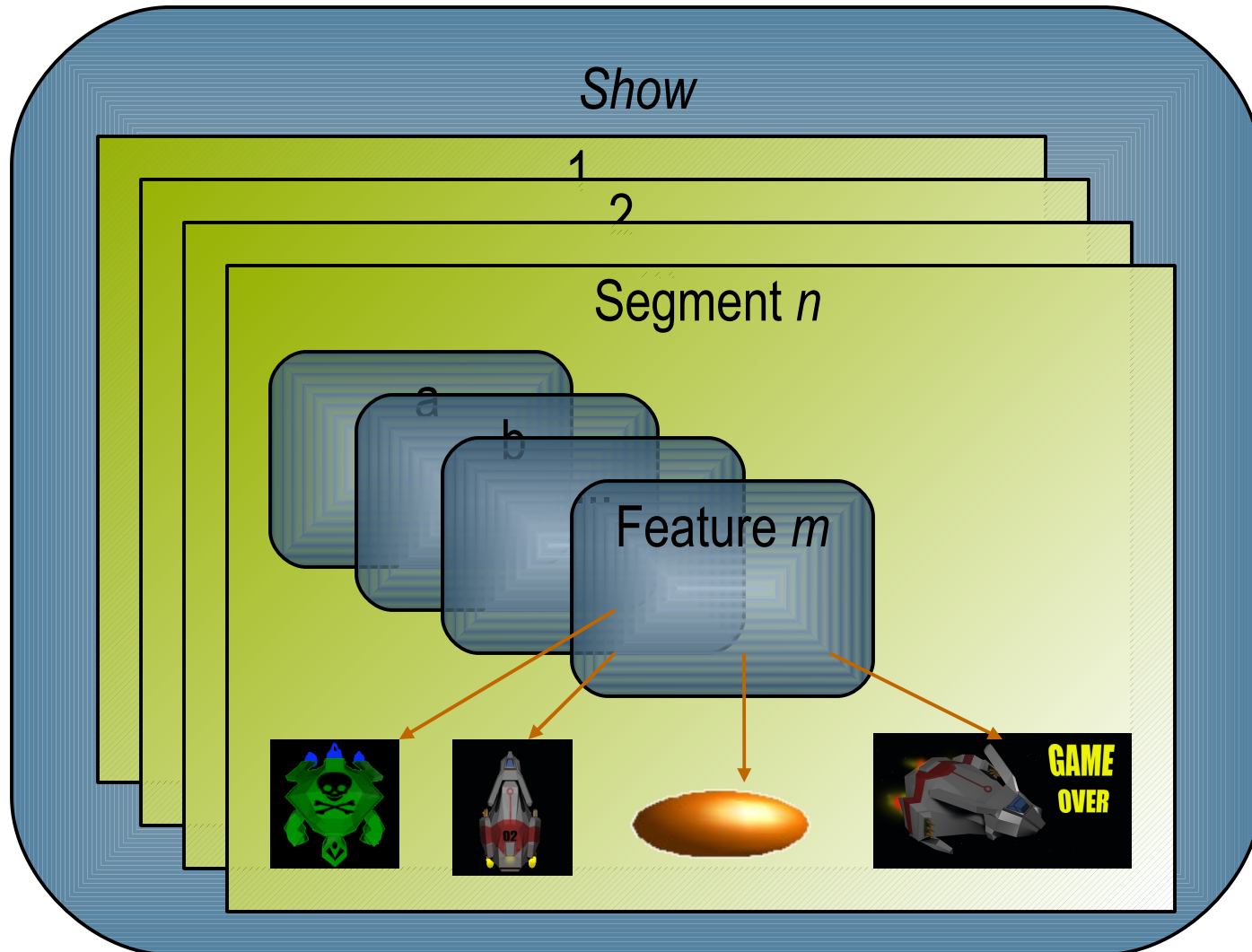
Game on!

Making a game with GRIN

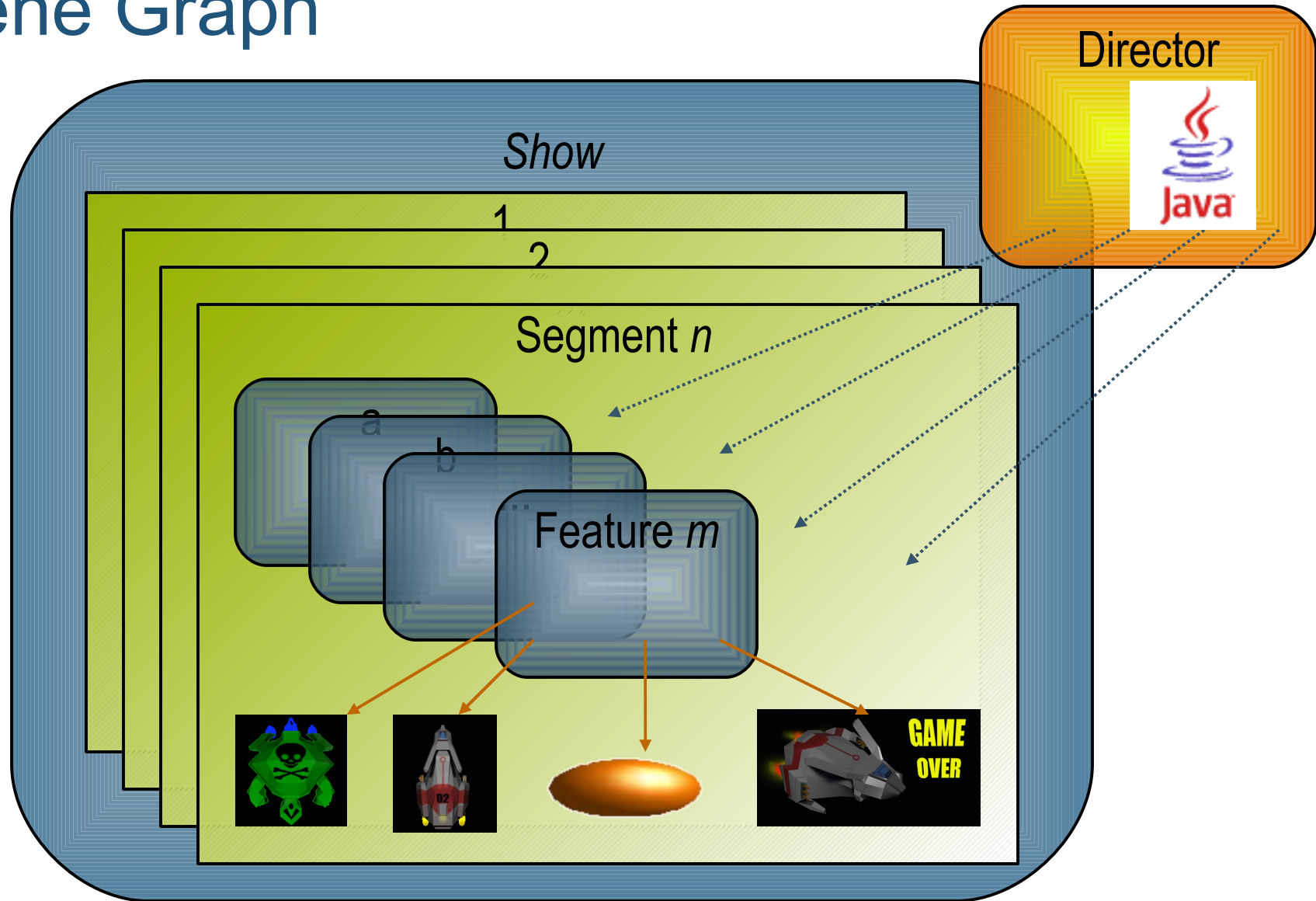
GRIN Framework

- > Scene Graph
- > The GRIN Elements

Scene Graph



Scene Graph



GRIN Elements

- > Show (scene graph)
 - Segments
 - Features
 - Different visual elements
 - commands
- > Show compiled to binary .grin file for runtime
 - MX Productions – GrinXML source form
- > Animation Loop (single threaded)
 - Calls your Director class, e.g. once per frame

Game Basics

- > Displaying Things
- > Moving objects
- > Receiving Key Presses
- > Controlling display (alpha, assembly, segment transition)
- > Cloning objects
- > Collision detection demo: Grinball
- > Game logic

Displaying Things

- > Single Segment
- > Image_Sequence
 - A running character
 - A jumping character
- > Assembly (shows one child part at a time)
 - Change a running character to jumping character
 - Change the digits on a score board
- > Translation
 - Move a character around the screen

Displaying Things: Define a Ball

```
<fixed_image id="fi.play.ball" x="0" y="0"  
    alignH="center" alignV="middle"  
    src="images/ball.png" />
```

```
<translator id="tr.play.ball"  
    translation="tn.play.ball">  
    <item feature="fi.play.ball"/>  
</translator>
```

```
<translation id="tn.play.ball" relative="false">  
    <keyframes>  
        <keyframe frame="0" x="0" y="0" />  
        <keyframe frame="1" x="0" y="0" />  
    </keyframes>  
</translation>
```

Displaying Things: Define the Game Segment

```
<segment id="sg.play">
  <active>
    <item feature="gr.play.background"/>
    <item feature="gr.play.bumpertop3"/>
    <item feature="gr.play.bumpertop2"/>
    <item feature="gr.play.bumpertop1"/>
    <item feature="tr.play.ball"/>
    <item feature="tmr.heartbeat"/>
  </active>
  <setup>
    <item feature="global.preloadGr"/>
  </setup>
</segment>
```

Moving GRIN features from Java

```
//Grin Features
```

```
private Feature tlrPinball;  
private InterpolatedModel imPinball;
```

```
//Director has access to Grin Features
```

```
    tlrPinball = director.getFeature("tr.play.ball");  
    imPinball =  
        (InterpolatedModel)director.getFeature("tn.ball");
```

```
//Director can get/set attributes of Features
```

```
public int getXPos() {  
    return imPinball.getField(Translator.X_FIELD);  
}
```

```
public void setXPos(int x) {  
    imPinball.setField(Translator.X_FIELD, x);  
}
```

Receiving Key Presses

> RC_Handler

- Fire events to the Director
- Arrow Keys
 - Key_Pressed
 - move character Translation
 - Activate animation Assembly
 - Key_Release (on *most* players)
 - stop character Translation
 - Activate standing still Assembly
- Enter Key
 - Fire Weapon

Controlling display: Alpha

> Fade

```
<fade id="fd.characterDeath"
  feature="fi.characterDeath">
  <keyframes>
    <keyframe alpha="255" frame="0"
interpolation="linear"/>
    <keyframe alpha="0" frame="5"
interpolation="linear"/>
  </keyframes>
<end_commands>
  <activate_segement segment="sg.gameOver"/>
</end_commands>
</fade>
```

Controlling display: Assembly

> Assembly

```
<assembly id="a.character" >
  <assembly_part id="p.stand" feature="g.stand" />
  <assembly_part id="p.runRight"
    feature="is.runRight"/>
  <assembly_part id="p.runLeft"
    feature="is.runLeft"/>
  <assembly_part id="p.jumpRight"
    feature="is.jumpRight"/>
  <assembly_part id="p.jumpLeft"
    feature="is.jumpLeft"/>
</assembly>
```

Controlling display: Segment Transition

```
<segment id="sg.play">
  <active>
    <item feature="gr.play.background"/>
    <item feature="gr.play.bumpertop3"/>
    <item feature="gr.play.bumpertop2"/>
    <item feature="gr.play.bumpertop1"/>
    <item feature="tr.play.ball"/>
    <item feature="tmr.heartbeat"/>
  </active>
  <setup>
    <item feature="global.preloadGr"/>
  </setup>
  <next>
    <activate_segment segment="sg.gameOver"/>
  </next>
</segment>
```

Cloning GRIN features from Java

```
HashMap clones = new HashMap();

this.tlrPinball =
    protoPinball.tlrPinball.cloneSubgraph(clones);

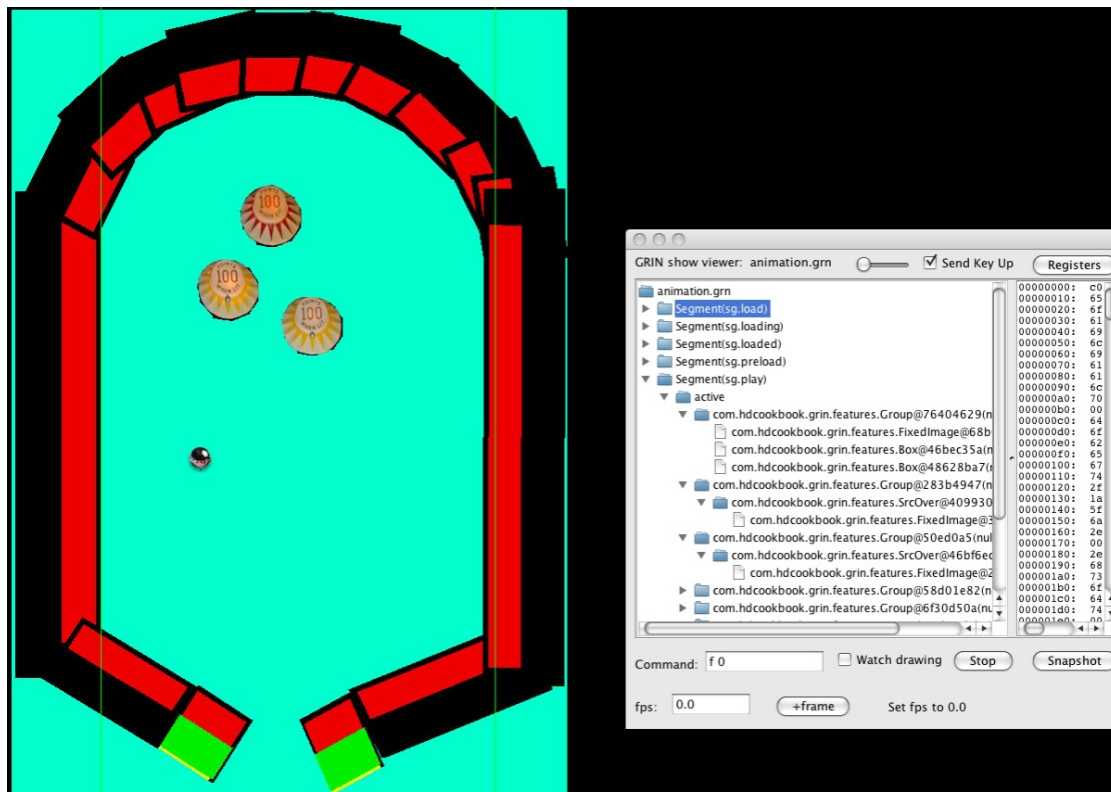
this.imPinball = (InterpolatedModel)
    clones.get(protoPinball.imPinball);

    // imPinball now has a clone of the node
    // protoPinball.tlrPinball, and all of its
    // child nodes.
```


Collision detection example: Grinball

Segments
Features
Fixed_Image
Translation
Rectangle

...



Game logic

```
public void heartbeat() {  
  
    // Move pinball  
    pinball[0].move();  
  
    CollisionPoint cp = collisionData.getPoint(x, y);  
  
    if (cp.type.equals("bumper")) {  
        boundary.setLength(15);  
    }  
  
    boundary.setAngle(cp.angle);  
    pinball[0].hitBoundary(boundary, cp);  
}
```

Game Over!

Measuring xlet performance

Performance Profiling

- > HD cookbook has a simple profiling mechanism
 - **`System.currentTimeMillis()`** is inadequate
 - Resolution can be in tens of milliseconds
 - Solution: Send UDP packets to a PC
 - Time with **`System.nanoTime()`**

```
// Set up profiler to send data to a PC at PROFILE_IP_ADDRESS

if (Debug.PROFILE && PROFILE_IP_ADDRESS != null) {
    Profile.initProfiler(2008, PROFILE_IP_ADDRESS);
}
```

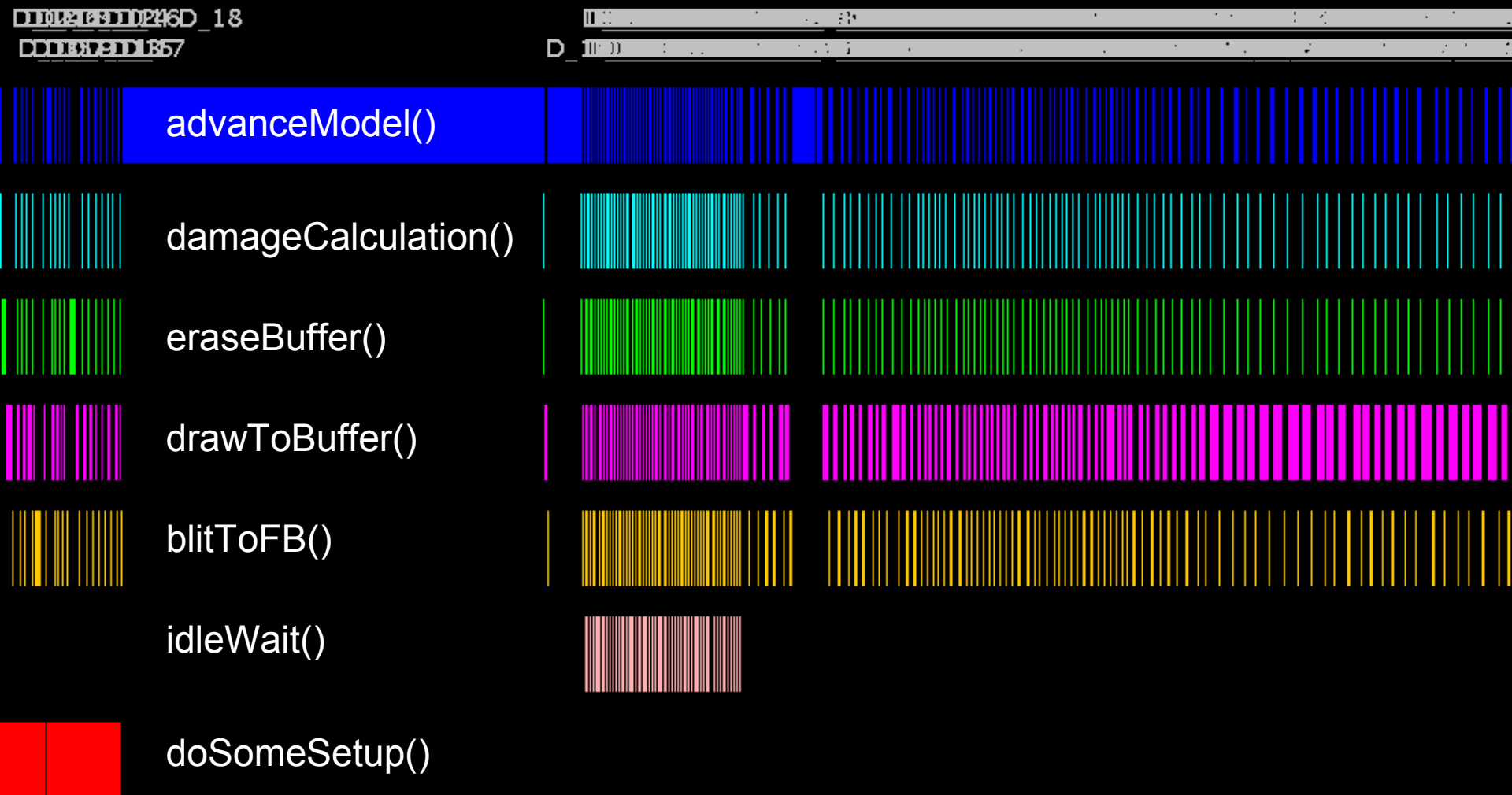
Performance Profiling

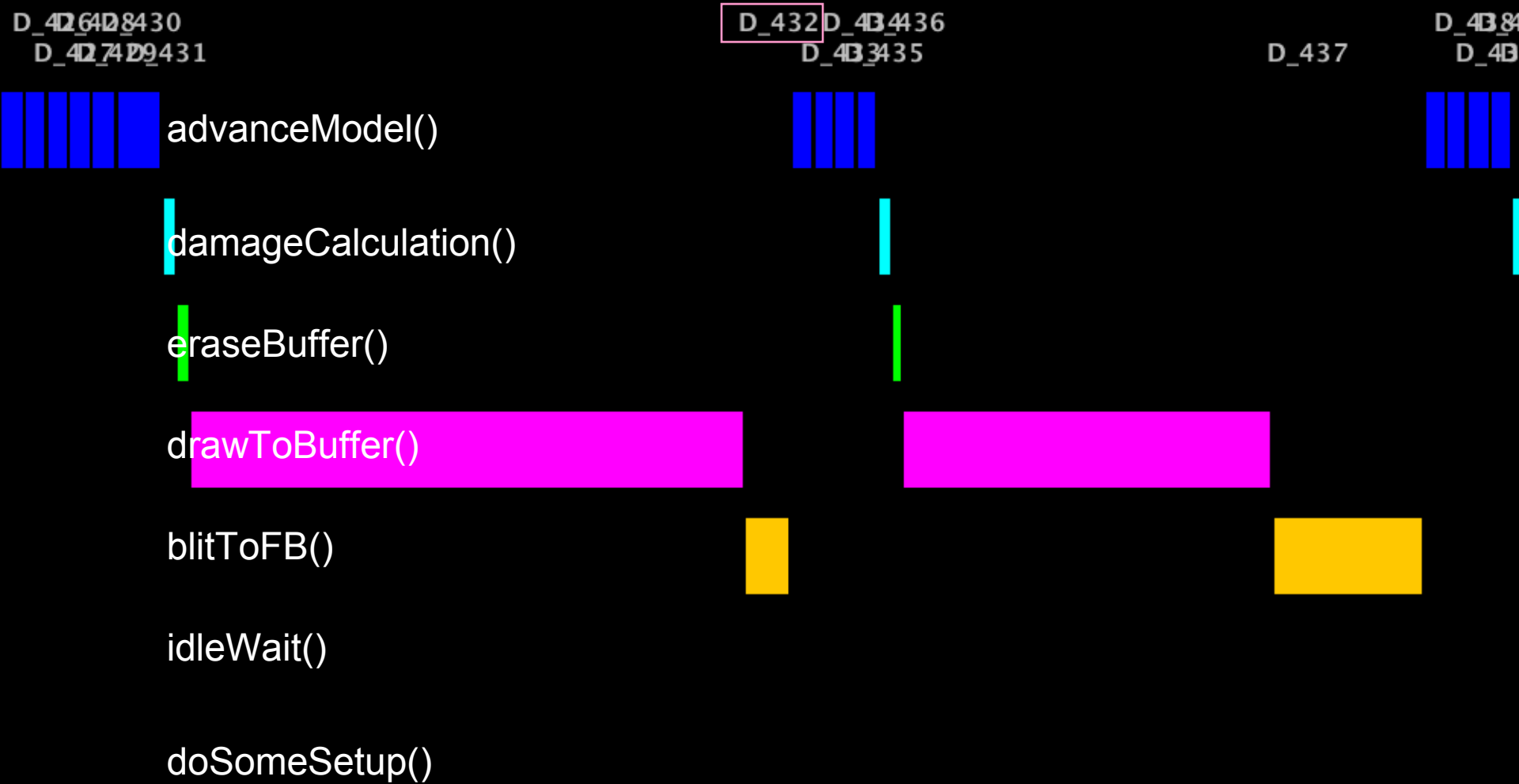
> Time important parts of xlet

```
private byte[] profileDraw;           // Time spent drawing to buffer

protected AnimationEngine() {
    if (Debug.PROFILE) {
        profileDraw = Profile.makeProfileTimer("drawToBuffer("+this+")");
    }
}

protected final void showFrame() throws InterruptedException {
    int tok2;
    if (Debug.PROFILE) {
        tok2 = Profile.startTimer(profileDraw, Profile.TID_ANIMATION);
    }
    callPaintTargets();
    if (Debug.PROFILE) {
        Profile.stopTimer(tok2);
    }
}
```





Profiling debug message

p432

```

00000000: 70 61 69 6e 74 54 61 72 67 65 74 73 20 6a 61 76 paintTargets jav
00000010: 61 2e 61 77 74 2e 52 65 63 74 61 6e 67 6c 65 5b a.awt.Rectangle[
00000020: 78 3d 33 31 36 2c 79 3d 37 32 31 2c 77 69 64 74 x=316,y=721,widt
00000030: 68 3d 31 32 39 38 2c 68 65 69 67 68 74 3d 32 32 h=1298,height=22
00000040: 39 5d 20 6a 61 76 61 2e 61 77 74 2e 52 65 63 74 9] java.awt.Rect
00000050: 61 6e 67 6c 65 5b 78 3d 36 32 30 2c 79 3d 34 31 angle[x=620,y=41
00000060: 35 2c 77 69 64 74 68 3d 33 34 32 2c 68 65 69 67 5,width=342,heig
00000070: 68 74 3d 32 35 39 5d 20 6a 61 76 61 2e 61 77 74 ht=259] java.awt
00000080: 2e 52 65 63 74 61 6e 67 6c 65 5b 78 3d 32 36 34 .Rectangle[x=264
00000090: 2c 79 3d 31 37 33 2c 77 69 64 74 68 3d 33 32 2c ,y=173,width=32,
000000a0: 68 65 69 67 68 74 3d 32 31 5d 20 6a 61 76 61 2e height=21] java.
000000b0: 61 77 74 2e 52 65 63 74 61 6e 67 6c 65 5b 78 3d awt.Rectangle[x=
000000c0: 31 31 30 31 2c 79 3d 35 34 35 2c 77 69 64 74 68 1101,y=545,width
000000d0: 3d 32 31 31 2c 68 65 69 67 68 74 3d 31 32 39 5d =211,height=129]

```

paintTargets

```

java.awt.Rectangle[x= 316, y= 721, width=1298, height= 229]
java.awt.Rectangle[x= 620, y= 415, width= 342, height= 259]
java.awt.Rectangle[x= 264, y= 173, width= 32, height= 21]
java.awt.Rectangle[x=1101, y= 545, width= 211, height= 129]

```


Summary

- > Tools are available at hdcookbook.com
 - click on HD cookbook open-source project

- > The same frameworks work for other Personal Basis Profile TV environments
 - MHP
 - Tru2way (OCAP)
 - IPTV

- > Java + scene graph = scripting environment



JavaOneSM

Thank You

Philip C. Starner
philip.starner@gmail.com

<http://www.mx-production.com/>

Bill Foote
bill.foote@sun.com

<http://jovial.com>

