

共有アプリで遊ぼう！

オープンソースカンファレンス
2011 Hiroshima

日本Red5ユーザー会
文屋 宏

アジェンダ

- はじめに
- Red5 の概要
- Flash Media Sever と Red5
- アプリ開発入門
- アプリを作ってみよう！！
- その他のデモ

はじめに

Red5 ユーザー会
公式ハッシュタグ
#red5ug

ぜひ今日の感想をツイートしてください！！

自己紹介

○氏名

文屋 宏(ぶんや ひろし) Twitter @bun_hiroshi

○所属

サーバホスティング会社

○業務

プロジェクトマネージャっぽいこと

○興味あること

分散データベース, クラウドコンピューティング, ウェブ系の技術

○活動

日本 Red5 ユーザー会メンバー

Red5 の概要

Red5 とは

■ Adobe Flash Media Server互換のメディアサーバ

⇒ RTMP(Real-Time Messaging Protocol)に準拠

■ オープンソースソフトウェア

⇒ ライセンスは LGPL

■ 開発言語は Java

⇒ Linux, Windows, Mac OSX上で動作可能

※ Red5本家サイト <http://www.red5.org/>

Red5 標準デモ

デモを見てみよう↓

<http://www.dog-lovers.jp:5080/demos/>

日本Red5ユーザー会

ユーザー会のサイト↓

<http://www.red5.gr.jp/>

Flash Media Server と Red5

Flash Media Sever とは？

■ Adobe製のサーバーソフトウェア

- ⇒ ビデオストリーミング
- ⇒ リアルタイムコミュニケーション

■ Flash Playerから接続可能

- ⇒ Flash Playerは全世界のPCで98%の普及率

■ 開発言語

- ⇒ Action Script で開発

■ エディションは3つ

- ⇒ Flash Media Streaming Server (12万円程度)
- ⇒ Flash Media Interactive Server (62万円程度)
- ⇒ Flash Media Enterprise Server (オープン価格)

Red5 で何ができるか

- ◆音声ファイルや動画ファイルのストリーミング配信
 - ◆Web カメラなどからのストリーミング映像を録画
 - ◆映像や音声などのライブ映像をリアルタイム配信
 - ◆複数のクライアント間でオブジェクトを共有
 - ◆いろんなライブラリを組み合わせで柔軟な開発
- ⇒実は FMS よりも自由度が高い！

アプリ開発入門

開発環境

◆クライアントサイド (ActionScript)

- Adobe Flash
- Adobe Flex
- OpenLaszlo (<http://laszlo.jp/>)
- FlashDevelop (<http://flashdevelop.jp/>)

◆サーバサイド (Java)

- Eclipse
- NetBeans

クライアントサイド入門

FlashDevelop を動かしてみよう！

<http://flashdevelop.jp/>

⇒ ActionScript3.0コンパイル入門

サーバサイド入門

Red5 の導入

◆Java のインストール

```
[root@sv]# yum install -y java
```

◆Red5 のダウンロード

```
[root@sv]# cd /opt
```

```
[root@sv opt]# wget http://wiki.red5.org/downloads/0_9/red5-0.9.1.tar.gz
```

```
[root@sv opt]# tar zxvf red5-0.9.1.tar.gz
```

◆Red5 の起動

```
[root@sv opt]# cd red5-0.9.1
```

```
[root@sv red5-0.9.1]# ./red5.sh > log/console_20111001.log 2 >&1 &
```

◆Red5 の停止

```
[root@sv red5-0.9.1]# ./red5-shutdown.sh
```

アプリを作ってみよう！！

寄せ書き(・・・というか落書き)アプリ

<http://demos.red5.gr.jp/rakugaki/>

⇒共有！！

描いた絵がリアルタイムに共有される！

例えばこちらは広島の人

例えばこちらは東京の人

今回、クライアントサイドの
開発は FlashDevelop

ソースコードを見てみよう

ボタンの作成

```
// 開始ボタンのラベルの作成
buttonText1 = new TextField();
buttonText1.text = "開始";
buttonText1.selectable = false;
buttonText1.width = 40;
buttonText1.height = 20;
buttonText1.x = 532;
buttonText1.y = 380;
addChild(buttonText1);
```

座標の指定は
ちょっと注意！

```
// 開始ボタンの作成
button1 = new SimpleButton();
button1.upState = makeRoundRect(0xDDDDDD, 90, 20, 10);
button1.overState = makeRoundRect(0xFFFFFFFF, 90, 20, 10);
button1.downState = makeRoundRect(0BBBBBB, 90, 20, 10);
button1.hitTestState = button1.upState;
button1.addEventListener(MouseEvent.CLICK, doClick1);
button1.x = 500;
button1.y = 380;
addChild(button1);
```

開始ボタンを押されたときの
イベント処理

開始ボタンを押されたときのイベント処理

```
//開始ボタンが押されたときのイベントハンドラ
private function doClick1(event:MouseEvent):void
{
    // マウス座標を読み取る
    stage.addEventListener(MouseEvent.CLICK, doMouseDown);
    stage.addEventListener(MouseEvent.CLICK, doMouseUp);
    stage.addEventListener(MouseEvent.CLICK, doMouseMove);
    text1.text = " 共有開始！！";

    rakugaki = new Sprite();
    addChild(rakugaki);
    nc = new NetConnection();
    nc.connect("rtmp://(ここで Red5 サーバを指定)/SO$ample");
    nc.addEventListener(NetStatusEvent.NET_STATUS, onNetStatus);
    rakugaki.addEventListener(MouseEvent.CLICK, startMove);
    rakugaki.addEventListener(MouseEvent.CLICK, stopMove);
    rakugaki.addEventListener(MouseEvent.CLICK, onMouseMove);
}
```

Red5 サーバに接続！



Red5 に接続したら、マウスの座標を同期

```
private function onNetStatus(evt:NetStatusEvent):void {
    switch(evt.info.code) {
        case "NetConnection.Connect.Success":
            so = SharedObject.getRemote("rakugaki", nc.uri, false);
            if ( so ) {
                so.addEventListener( SyncEvent.SYNC , function( evt:SyncEvent ):void {
                    if ( evt.target.data["rakugakiPosition"] ) {
                        rakugaki.x = so.data.rakugakiPosition.x;
                        rakugaki.y = so.data.rakugakiPosition.y;
                    }
                    so.setProperty("rakugakiPosition", { x: rakugaki.x, y: rakugaki.y } );
                    if ( evt.target.data["linedata"] ) {
                        linedata = so.data.linedata.data;
                    }
                    so.setProperty("linedata", {data: linedata} );
                });
                so.connect(nc);
            }
            break;
        case "NetConnection.Connect.Closed":
            break;
        case "NetConnection.Connect.Failed":
            break;
        case "NetConnection.Connect.Rejected":
            break;
        default:
    }
}
```


マウスをドラッグしたラインの描画

```
// ラインの描画
private function doEnter(Event):void {
    var wx:Number;
    var wy:Number;
    if (0 < linedata.length) {
        // 青い線で描画していく
        canvas_mc.graphics.lineStyle(3, 0x0000FF);
        wx = linedata[0][0] ;
        wy = linedata[0][1] ;
        canvas_mc.graphics.moveTo(wx, wy);
        for (var i:int = 1; i < linedata.length; i++) {
            wx = linedata[i][0] ;
            wy = linedata[i][1] ;
            canvas_mc.graphics.lineTo(wx, wy);
        }
    }
}
```

何をサーバにアップロードするか

binの中身をアップ

```
<meta name="keywords" content="" />

<script src="js/swfobject.js" type="text/javascript"></script>
<script type="text/javascript">
  var flashvars = {
  };
  var params = {
    menu: "false",
    scale: "noScale",
    allowFullscreen: "true",
    allowScriptAccess: "always",
    bgcolor: "#FFFFFF"
  };
  var attributes = {
    id:"rakugaki"
  };
  swfobject.embedSWF("rakugaki.swf", "altContent", "100%", "100%", "10.0.0", "expressInstall.swf"
</script>
<style type="text/css">
  html, body { height:100%; overflow:hidden; }
  body { margin:0; }
</style>
</head>
<body>
  <div id="altContent">
    <h1>rakugaki</h1>
    <p>Alternative content</p>
    <p><a href="http://www.adobe.com/go/getflashplayer">

# 参考文献

◆基本からしっかりわかるActionScript 3.0

<http://www.amazon.co.jp/dp/4839930252/>

◆ActionScript3.0開発のためのFlashDevelop完全ガイド

<http://www.amazon.co.jp/dp/4774143081/>

◆ Think IT 連載記事(第3回 はじめよう！Red5プログラミング)

<http://thinkit.co.jp/article/152/3>

◆その他ウェブの情報

では、数々のおもしろいアプリを  
ご覧ください！

