

Flash 5 动作脚本语言在交互性 开放性课件制作中的应用

Application of Flash 5 ActionScript to Interactive and Opening Courseware Making

莫治雄

Mo Zhixiong

(广西职业技术学院 南宁 530227)
(Guangxi Vocational College, Nanning, 530227)

摘要 以多项选择题课件设计为例,介绍交互性和开放性课件制作有关的Flash 5脚本语言元素,探讨应用Flash 5动作脚本语言制作交互性和开放性课件的方法。

关键词 Flash 动作脚本语言 多项选择题 课件

中图分类号 TP391.41

Abstract The elements of Flash 5 ActionScript concerning with interactive and opening courseware making are introduced. The design of multiple-choice test courseware is used as a sample. The method for making interactive and opening courseware with Flash 5 ActionScript is described.

Key words Flash, ActionScript, multiple-choice test, courseware

在互联网技术飞速发展的今天,通过网络实施教学已经成为重要的教学手段。课件上网实现资源共享和合作交流,是众多教师的共同愿望。

Flash 5 是一款矢量动画制作软件,应用Flash的动作脚本语言,可以制作出具有良好交互性和开放性的课件。本文以制作多项选择题课件为例,探讨应用Flash 5动作脚本语言制作交互性和开放性课件的方法。

1 与交互性开放性课件制作有关的Flash 5脚本语言元素

Flash动作脚本语言像其它脚本语言一样具有丰富的语言元素,包括常量、变量、运算符、表达式、函数、属性、动作、对象和电影剪辑等等。与创建交互性和开放性课件密切相关的有按钮、电影剪辑和相应的事件和动作。

1.1 鼠标事件管理动作与鼠标事件

Flash动作脚本语言中的On Mouse Event动作用于检测鼠标事件和键盘按键事件。本动作只能分配给按钮。当指定的鼠标或键盘按键事件发生时,执行该动作内的语句。该动作的

语法格式如下：

```
on (mouseEvent) {
    statements;
}
```

其中，mouseEvent 是指鼠标事件或按键事件，Flash 定义了以下 mouseEvent：

Press：鼠标指针在按钮上时按下鼠标按键；

Release：鼠标指针在按钮上时释放鼠标按键；

RollOver：鼠标指针移到按钮上面；

RollOut：鼠标指针从按钮上移出；

Release Outside：鼠标指针在按钮上时按下鼠标按键，移出按钮外后才释放按键；

DragOut：鼠标指针在按钮上时按下鼠标键，然后拖出按钮外；

DragOver：鼠标指针在按钮上时按下鼠标键，然后拖出按钮外，接着又拖回按钮上；

KeyPress：按下指定的键盘键（键盘键用键码或键常数指定，请参看 Flash 的帮助）。

这些鼠标事件可以在动作面板的参数区选择，且可以选择一个，也可以同时选择多个。

例如，可以给多项选择测试课件中的“下一题”按钮分配下面的动作：

```
on (release) {
    if (answer == correctanswer) {
        score = score+1;
    }
    gotoAndPlay (" 载入");
}
```

这段代码的功能是：当学生按下“下一题”按钮时，首先判断学生的选择答案是否正确。如果正确，分数累加 1 分，否则不加分。然后跳转“载入”帧，继续测试下一题。

1.2 电影剪辑事件管理动作与电影剪辑事件

Flash 动作脚本语言中的 onClipEvent 动作用于处理电影剪辑事件。该动作的语法格式如下：

```
onClipEvent (movieEvent); {
    statements;
}
```

动作中的 movieEvent 参数指定电影剪辑事件。本动作只能分配给电影剪辑实例。当指定的剪辑事件发生时，执行该动作内的语句。电影剪辑事件包括：

load：电影剪辑实例化并在时间轴中出现时触发本动作。

unload：在电影剪辑实例从时间轴中卸载之后，在第一帧触发本动作。与 unload 剪辑事件关联的动作在此帧中其它任何动作之前被处理。

enterFrame：在播放电影剪辑实例的任一帧时触发本动作。与 enterFrame 剪辑事件关联的动作在该帧中其它任何动作之后被处理。

mouseMove：每当移动鼠标时触发本动作。使用 _xmouse 和 _ymouse 属性可以确定当前鼠标的坐标。

mouseDown：鼠标左键按下时触发本动作。

mouseUp：释放鼠标左键时触发本动作。

keyDown: 当某一键被按下时触发本动作。使用 Key.getCode 方法可以获得最近一次按键的信息。

keyUp: 释放某一按键时触发本动作。

data: 当接收到 loadVariables 或 loadMovie 动作载人的数据时触发本动作。对于 load-Variables 动作, data 事件只在载入最后一个变量时发生 1 次; 对于 loadMovie 动作, 每读入一段数据 data 事件就发生 1 次。

这些剪辑事件可以在动作面板的参数区选择, 且每次只能选择 1 个。

例如, 可以给一个电影剪辑分配下面的动作:

```
onClipEvent (load) {  
    Mouse.hide ();  
    startDrag ("", true);  
}
```

这段代码的功能是: 当指定的电影剪辑载入时, 先隐藏标准鼠标光标, 然后允许把已载入的电影剪辑当作鼠标光标拖动。

1.3 载入外部文件的动作

Flash 动作脚本语言中的 loadVariables 动作的作用是从外部文件中载入变量和设置变量的值。该动作的语法格式如下:

```
loadVariables (url, location [, variables]);
```

动作中的参数 url 是载入外部文件的绝对或相对 URL。URL 中指向的主机必须与下载的 Flash 课件处在同一子域; location 是一个可选的参数, 用于指定接收变量的层级或目标。variables 是一个可选的参数, 用于指定发送变量的方法。

loadVariables 动作可以从外部文件, 如文本文件或 CGI 脚本、ASP (Active Server Pages)、PHP (Personal Home Page) 生成的文本中载入变量, 并在课件或电影剪辑中设置这些变量的值。loadVariables 动作还可以用于更新当前正在播放的课件中的变量的值。

在 loadVariables 动作中 url 参数指定的文本必须是标准的 MIME 格式。在文本中可以指定任意个变量。例如, 在我们的多项测试课件中, 存储测试信息的文本文件 test.txt 中有下面的内容, 定义了 3 个变量:

```
title=Flash5 基础多项选择测试 &.totalQuestions=4&.eof=1
```

可以用 loadVariables 动作把这个文本文件载入课件中:

```
loadVariables (" test.txt", 0);
```

这个动作的作用就是载入测试信息。载入这个文本文件后, 在课件中将设置 title、totalQuestions 和 eof 等 3 个变量及其取值。

2 多项选择题课件设计思路和算法

多项选择题课件的设计思路和算法如图 1 所示。

3 多项选择题课件实现方法

制作具有良好交互性和开放性的多项选择测试课件的关键技术是给按钮分配动作脚本和使用 loadVariables 动作载入试题文件。

3.1 试题文件

本课件采用数据与程序分开技术，把测试信息和试题内容存放在文本文件中，然后用 loadVariables 动作载入这些文本文件。本课件使用了下列 5 个文本文件：

- test.txt 定义测试标题和题目总数
- question1.txt 第 1 题的题目及答案
- question2.txt 第 2 题的题目及答案
- question3.txt 第 3 题的题目及答案
- question4.txt 第 4 题的题目及答案

使用外部文本文件可以轻易实现课件的开放性。因为测试的课程和试题内容均存放在外部文件中，只要熟悉文本编辑的教师均可以按照自己的课程要求设计自己的多项选择题。

文本文件的书写要求是：变量名=值（如：title=多项选择测试），变量/值对之间用“&”符号分隔，值内单词之间用“+”分隔。变量定义建议在一个逻辑行中连续写完。例如题目文本文件 question2.txt 的内容如下：

q=根据 Flash+5 的变量命名规则，下列变量名中哪个是不正的确？&answer1=for
&answer2=the&answer3=you&answer4=she&correctanswer=1&eof=1

变量 q 为题目，answer1~answer4 是 4 个答案，correctanswer 是正确答案，eof 是判断文件结束的标志变量。

如果要换行，也可以写成如下形式：

q=根据 Flash+5 的变量命名规则，下列变量名中哪个是不正确的？
 &answer1=for
 &answer2=the
 &answer3=you
 &answer4=she
 &correctanswer=1&eof=1

3.2 变量说明

在动作脚本中使用了下面的变量：

- totalQuestion 题目总数，在 test.txt 中定义
- correctanswer 正确答案序号，在题目文本文件（如 question1.txt）中定义
- eof 标志变量，用于判断题目读入是否已经结束，0-没有结束，1-读入结束
- qnumber 题目序号（如 1，2，3，...）
- question 题目字符串（序号+题目）
- answer 学生选择的答案序号（1、2、3 或 4）
- view 是否查看正确答案的标志变量，0-没有查看，1-查看
- score 累计测试得分（每题 1 分）

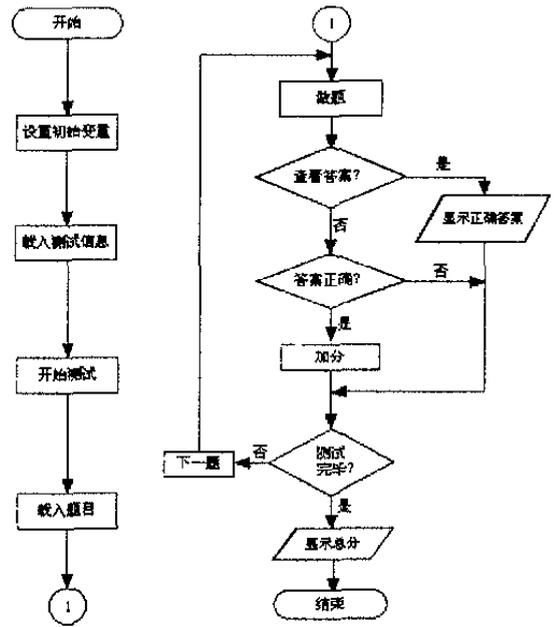


图 1 多项选择题课件设计思路和算法

myscore 测试总得分 (百分制)

curQuestions 当前答题序号字符串, 如: 第 2 题 (共 4 题)

output 保存用于显示的正确答案序号的字符串, 如: 正确答案是: 2

3.3 设计说明

多项选择测试课件主时间轴的设计见图 2。

本课件把主时间轴中划分为开始、设置、信息、测试、载入、循环、已载入和完成等帧, 各帧的功能说明如下:

3.3.1 开始

“开始”帧测试课件是否已经下载完毕, 如果没有下载完毕, 在此帧循环播放。如果下载完毕, 转到“设置”帧播放。相应的动作脚本是:

```
if (_framesloaded >= _totalframes) {
    gotoAndPlay (" 设置");
}
```

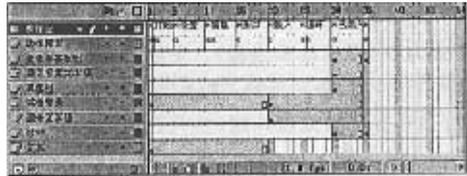


图 2 多项选择测试设计

3.3.2 设置

“设置”帧的作用是初始化变量, 载入测试信息 (本次测试的标题和题目总数), 动作脚本如下:

```
score=0;
view=0;
qnumber=0;
eof=0;
loadVariables (" test.txt", "", " POST");
```

在文本文件 test.txt 中, 设置了 3 个变量 title、totalQuestions、eof。title 变量用来存储测试标题, 如“Flash5 基础多项选择测试”; totalQuestions 存储题目总数; eof 标志本文件结束。内容如下:

```
title=Flash5 基础多项选择测试 &totalQuestions=4&eof=1
```

3.3.3 信息

“信息”帧的作用是测试所需的测试信息 (即 test.txt) 是否已经载入。如果没有载入完毕, 在此处循环等待; 如果已经载入, 则转到“测试”帧。动作脚本如下:

```
if (eof==1) {
    gotoAndStop (" 测试");
}
```

3.3.4 测试

“测试”帧等待学生单击“开始”按钮, 然后跳转“载入”帧播放。课件中给“开始”按钮添加了以下脚本:

```
on (release) {
    gotoAndPlay (" 载入");
}
```

3.3.5 载入

“载入”帧的作用是载入试题文本文件 (如 question1.txt)。此处的帧动作如下:

```
eof=0;
```

```

answer==0;
qnumber=qnumber+1;
if (qnumber>totalQuestions) {
    gotoAndStop (" 完成");
} else {
    curQuestions=" 第 " +qnumber+" 题" +" (共 " +totalQuestions+" 题)";
    loadVariables (" question" +qnumber+" .txt", "", " POST");
}

```

在这段脚本中,首先设置载入结束标志变量 eof 为 0(载入没有结束),学生选择的答案 answer 为 0(表示尚未选择)。变量 qnumber 确定载入题目的文本文件的序号(如 question1.txt 中的 1)。判断语句判断,如果文本文件的序号大于题目总数,跳转“完成”帧,结束测试。否则,显示现在测试第几题、共几题的信息,载入 qnumber 确定的题目文本文件。例如,当 qnumber=3 时,载入 question3.txt。

3.3.6 循环

“循环”帧的作用是等待载入题目文本文件。如果没有载入完毕,就在此处循环播放等待载入;如果已经载入完毕,则跳转“已载入”帧,进行当前题目的测试。本帧动作脚本如下:

```

if (eof==1) {
    gotoAndPlay (" 已载入");
}

```

3.3.7 已载入

“已载入”帧的作用是显示题目供学生作答。这里的帧动作脚本如下:

```

question=qnumber + " ." + q;
stop ();

```

第一个语句的作用是构造题目(如:2. 根据 Flash 5 的变量命名规则,下列变量名中哪个是不正确的?),然后在动态文本框 question

中显示出来。4 个答案 answer1~answer4 也同时在相应的动态文本框中显示。stop() 语句的作用是等待学生选择答案。播放课件时的画面如图 3 所示。

在答题画面(图 3)中,学生可以单击答案前面的单选按钮,作出选择。课件中均给这些单选按钮电影剪辑内的按钮分配了脚本。以答案 1 为例,脚本如下:

```

on (release) {
    gotoAndStop (2);
    _parent. answer=1;
}

```

_parent. answer=1 语句记录了学生选择的是第 1 个答案。

当学生选择完毕,可以单击“下一题”按钮,继续做下一题的测试。课件中给“下一题”按钮分配了以下脚本:

```

on (release) {
    if (answer==correctanswer && view == 0) {

```



图 3 多项选择测试界面

```
        score=score+1;
    }
    output="";
    view=0;
    gotoAndPlay (" 载入");
}
```

这段代码的功能是:当学生按下本按钮时,首先判断学生的选择答案是否正确。如果正确,且没有查看过正确答案($view=0$),则分数累加1分。否则不加分。然后跳转“载入”帧,继续测试下一题。

学生对当前题目没有把握,也可以单击“查看答案”按钮,查看正确答案。课件中给“查看答案”按钮分配了以下动作脚本:

```
on (release) {
    output=" 正确答案是:" + correctanswer;
    view=1;
}
```

当学生单击本按钮时,即把正确答案的序号显示在 output 动态文本框中,然后记录其已经查看了正确答案($view = 1$)。这样,单击“下一题”按钮时,即使学生答对了也不会加分。单击“查看答案”按钮,查看正确答案,起到即时反馈的作用。如果要课件更加完善,充分体现交互性的作用,可以在这里显示解答此题需要掌握的相关知识和分析,使学生得到及时辅导。

如果学生测试到中途不想继续测试,可以单击“终止测试”按钮,跳转到“完成”帧播放。分配给“终止测试”按钮的动作脚本如下:

```
on (release) {
    if (answer==correctanswer && view == 0) {
        score=score+1;
    }
    output="";
    view=0;
    gotoAndPlay (" 完成");
}
```

3.3.8 完成

“完成”帧的功能是显示学生测试所得的分数(百分制)。这一帧的动作脚本如下:

```
myscore=int ((score/totalQuestions) * 100);
curQuestions="";
stop ();
```

$myscore=int ((score/totalQuestions) * 100)$ 语句的作用是把分数换算为百分制。这一帧画面中还有一个“重新测试”按钮,单击该按钮可以重新进行测试。

4 结语

应用 Flash 的动作脚本语言制作课件,采用的是数据与程序分开技术,教师可以根据自己的教学需要定制自己的测试内容。开放性课件的制作有利于课件共享和推广应用。

Flash 使用矢量技术制作课件,生成文件的字节少,在网上采用流式播放技术播放,可以边下载边播放,降低了对带宽的要求,“减少”了等待时间,有利于课件的网上教学和网上交流。

使用 Flash 制作课件还可以制作很多图符(如按钮、图形、电影剪辑),作为积件存放在 Flash 的图符库中,通过逐步积累,可以构建课件资源库,让广大教师共享,提高课件的制作水平和开发效率。

本课件还有一些不足的地方,比如:可以加上“上一题”按钮,测试时间倒计时,测试开始和结束时响铃等等功能;本课件要求每一道题目建立一个文本文件,如果题目很多要创建的文本文件也就很多。要改进这些不足,需要把题目和相应的答案存放在数据库中,然后通过 ASP 脚本处理数据库的数据。关于这个方面的实现方法,作者将另拟文探讨。

(责任编辑:邓大玉)

(上接第 259 页)

```

Else
    lh1=lh1-0.001
End If
Loop
wks.Cells(18,2)=lh1
wks.Cells(19,2)=q
wks.Cells(20,2)=Sqr(lh1*lh1-2*q/k1*L)
End Sub

```

每次输入数据完毕,按一次计算按钮,即得出结果,这些原始数据和计算结果可以复制到工程设计文档中。

4 结语

采用电子表格软件进行工程设计计算,方便快捷,对尚未有现成计算软件的情况,这是一个很好的解决办法。

参考文献

- 1 土坝设计.上册.北京:水利电力出版社,1978.
- 2 潘家铮.建筑物的抗滑稳定和滑坡分析.北京:水利出版社,1980.

(责任编辑:蒋汉明)