Making of ArcadeEngine

by Malte Brill

Once I started with LiveCode I was looking for a more affordable alternative to Macromedias Director. I just had finished Multimedia Highschool and had my diploma as Mediengestalter Bild und Ton (Mediadesigner for Video and Audio) from Hamburgs Chamber of Commerce in my pocket. Setting up my one man company I wanted to create Multimedia CD-Roms as advertising medium and thought I needed an authoring system that allowed me to build nice menus around my movies.
Looking on versiontracker for an alternative to Director I discovered the Revolution 1.1.1 Starter Kit and became hooked. With the help of the use-list I managed to solve the problems I had on my learning curve. I was in touch with one of the biggest german Macintosh magazines at that time and they asked me to write a review and workshop on using Revolution and that's how the story started.
I thought that I should test more of LiveCode's multimedia abilities, as I needed to compare features directly to other autjoring systems, emphasis on the comparison with Director. So I created a few simple animations. Soon I discovered that the animation manager that came with LiveCode by that time had its weaknesses. So I tried scripting my animations and came up with the first versions of circular movement. I experimented a lot and soon I found it was a good idea to poduce reusable code.
The first version worked pretty well for moving things but was too clumsy to be used by other people than me. I produced a game for a recycling company using the library and it worked well. But it was not convenient enough to share it with other developers. All functions required numerical input to produce meaningful results. I used to explicitly pass parameters to the functions I used, so if I wanted to do something with two controls locations, I needed to split those locations into their x and y coordinates in the calling script. A calling script snipped looked like this:

```
put item 1 of the loc of btn "myButton1" into x1
put item 2 of the loc of btn "myButton1" into y1
put item 1 of the loc of btn "myButton2" into x2
put item 2 of the loc of btn "myButton2" into y2
get distance(x1,y1,x2,y2)
```

The distance function looked like this:

```
function distance x1,y1,x2,y2
  put x1-x2 into oppositeLeg
  put y1-y2 into adjacentLeg
  put round(sqrt (oppositeLeg^2+adjacentLeg^2)) into hypotenuse
  return hypotenuse
end distance
```

This worked, but required a lot of scripting in the stack. What I wanted to be able to do was a single function call like this:

```
get distance(the loc of btn "myButton1",the loc of btn "myButton2")
```

So I needed to revise all the functions I had. That meant a few more lines of script in the library, but ease of use when working with it. The same function, including error checking for the parameters now looks like this:

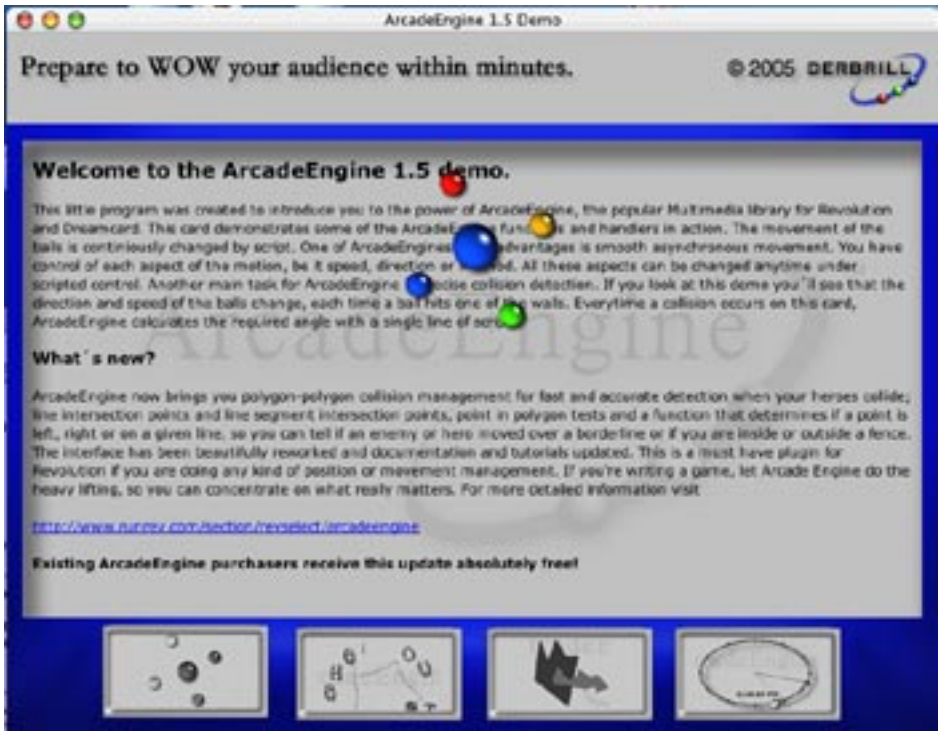
```
function distance
  repeat with i=1 to paramcount()
    if i<paramcount() then
      put param(i)&"," after theValue
    else
      put param(i) after theValue
    end if
  end repeat
  repeat with i=1 to the number of items of theValue
    if item i of theValue is not a number then
      return "Error: All Parameters must be numbers!"
      exit distance
      exit repeat
    end if
  end repeat
  put item 1 of theValue into x1
  put item 2 of theValue into y1
  put item 3 of theValue into x2
  put item 4 of theValue into y2
  if y2 is empty then
    return "Error: Syntax is distance(x1,y1,x2,y2)"&cr&theValue
  end if
  put x1-x2 into oppositeLeg
  put y1-y2 into adjacentLeg
  put round(sqrt (oppositeLeg^2+adjacentLeg^2)) into hypotenuse
  return hypotenuse
end distance
```

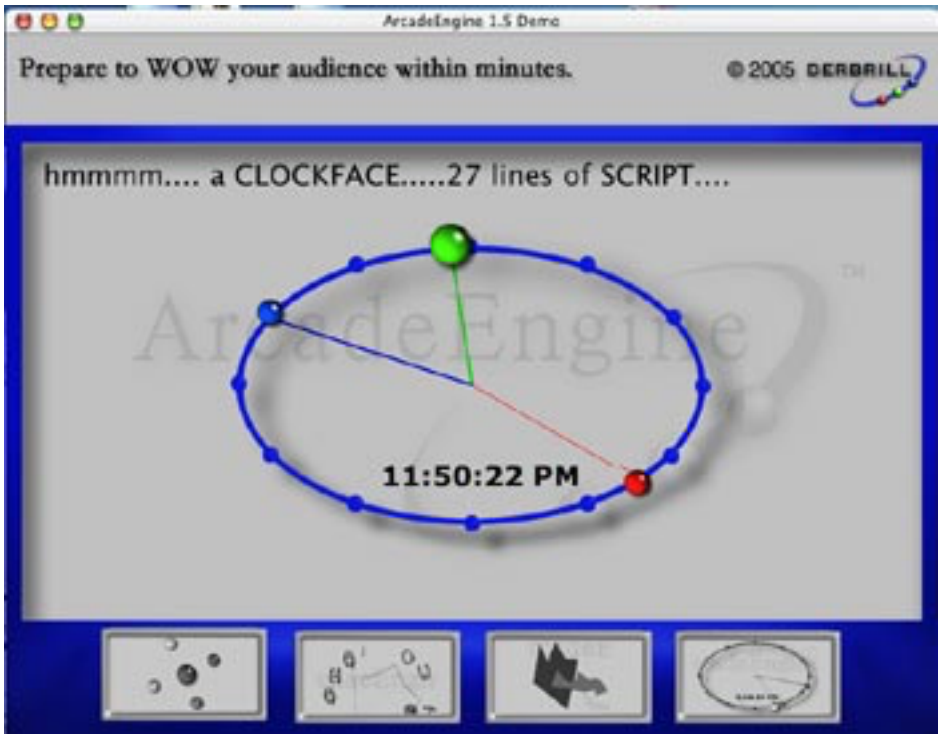
Now the function allowed both numerical and easy input which is eactly what I wanted it to.

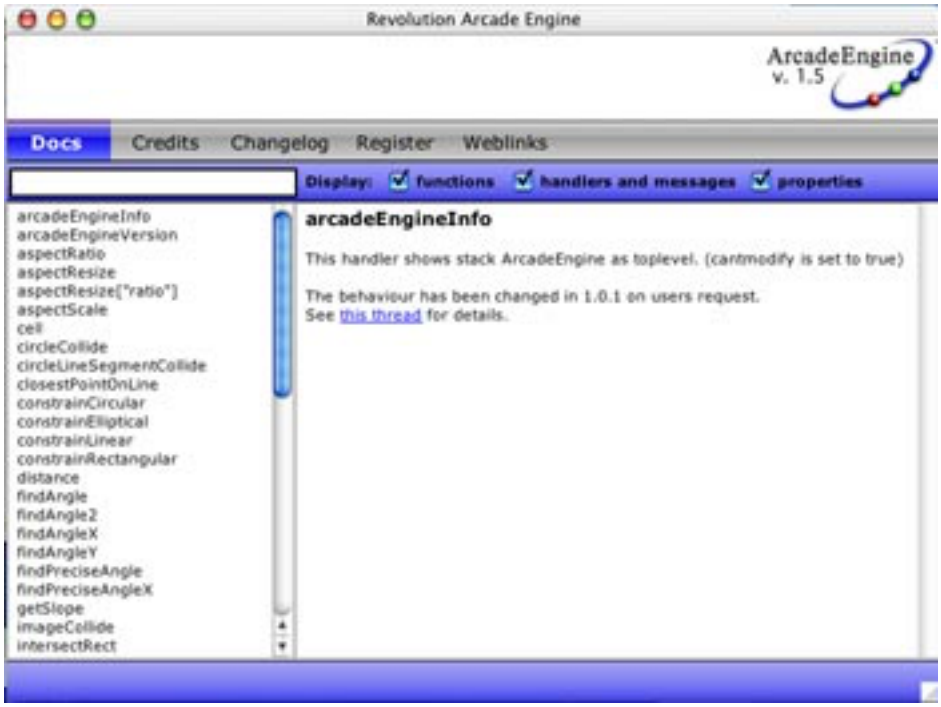
My little library now had some functions and handlers to move controls and also the first basic collision detection routine. I produced another game for the recycling company which was well received. I was asked to talk on the EuroRevCon on Malta. So I put together a few simple demo stacks and met many nice people there. I had the chance to talk to Kevin Miller about the things I do with LiveCode and we discussed the furture of my library in a later meeting in Edinburgh.

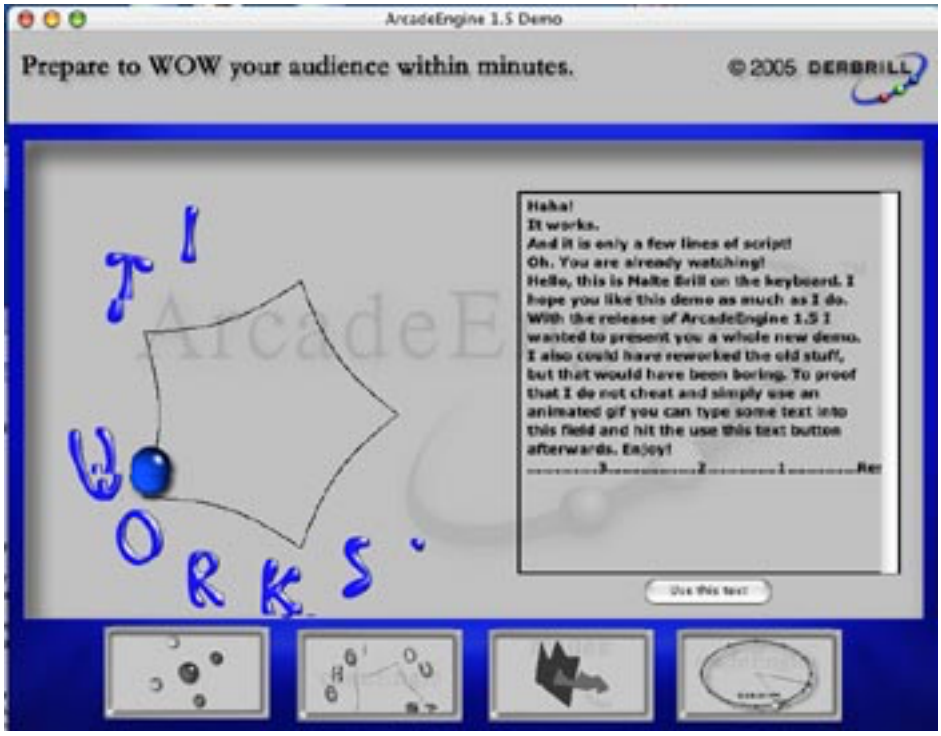
It soon became clear that it had the potential to help others creating games and animations. But still there were a few things missing. I added pixel precise collision detection for images, some nice path creation routines and a few more geometric functions.
I concentrated on optimizing existing handlers and functions for speed. My goal was to make everything run smoothly on my oldest machine, a G4-400 MHz. And it needed to be as stable as possible.

There were quite a few people helping me a lot throughout development with constructive critisism, suggestions and kicks in the butt when I needed them. The 1.0 version was very well received. After taking a few days off I looked at the documentation again and thought: "Hey, it is a multimedia library, but it looks horrible!" So I invested a few days in a new documentation system and added a few functions for point in polygon tests and polygon polygon collisions.




ArcadeEngine 1.5 Demo
Prepare to WOW your audience within minutes.
© 2005 DERBRILL
hmmmm.... a CLOCKFACE.....27 lines of SCRIPT....
11:50:22 PM


Revolution Arcade Engine
ArcadeEngine
v. 1.5
Docs
Credits
Changelog
Register
Weblinks
Display:
functions
handlers and messages
properties
arcadeEngineInfo
arcadeEngineVersion
aspectRatio
aspectResize
aspectResize["ratio"]
aspectScale
cell
circleCollide
circleLineSegmentCollide
closestPointOnLine
constrainCircular
constrainElliptical
constrainLinear
constrainRectangular
distance
findAngle
findAngle2
findAngleX
findAngleY
findPreciseAngle
findPreciseAngleX
getSlope
imageCollide
intersectRect
arcadeEngineInfo
This handler shows stack ArcadeEngine as toplevel. (cantmodify is set to true)
The behaviour has been changed in 1.0.1 on users request.
See this thread for details.


ArcadeEngine 1.5 Demo
Prepare to WOW your audience within minutes.
© 2005 DERBRILL
Haha!
It works.
And it is only a few lines of script!
Oh. You are already watching!
Hello, this is Malte Brill on the keyboard. I hope you like this demo as much as I do.
With the release of ArcadeEngine 1.5 I wanted to present you a whole new demo.
I also could have reworked the old stuff, but that would have been boring. To proof that I do not cheat and simply use an animated gif you can type some text into this field and hit the use this text button afterwards. Enjoy!
3 2 1 Re
Use this text