

Designing a Scripting Language to Help the Blind Program Visually

Kenneth G. Franqueiro and Robert M. Siegfried
Department of Mathematics and Computer Science
Adelphi University, Garden City, NY 11530 USA
(516)877-4482, {franquek, siegfri}@panther.adelphi.edu

ABSTRACT

The vast proliferation of GUI-based applications, including graphical interactive development environments (IDEs), has placed blind programmers at a severe disadvantage in a profession that had previously been relatively accessible. Visual Basic is one such programming language and IDE, in which most programmers “point and click” to design the forms on which their applications rely. It is the goal of this project to introduce a scripting language that eliminates this barrier by providing a scripting language that makes it possible to define Visual Basic GUI forms and their components verbally, while remaining easy to write.

Categories and Subject Descriptors

K.3.2 [Computers and Education]: Computer and Information Science Education - *Computer science education*.

K.4.2 [Computers and Society]: Social Issues - *Assistive technologies for persons with disabilities*.

General Terms

Human Factors, Languages

Keywords

Visual Basic, Visual Programming, Graphical User Interfaces, Blind Programmers.

INTRODUCTION

Since the 1960s, the blind have studied computer programming and have been able to secure employment as professional programmers. For example, the University of Manitoba ran a training program for blind programmers that started in the mid-1960s and ran for many years that had a low attrition rate and high job placement rate[4]. The American Federation for the Blind had 130 blind programmers in their database as of 1998[1].

While there have been many efforts to make the World Wide Web[3][7] and visual representations of data[2] more accessible to the blind, there have been fewer efforts to make it easier for the blind to program beyond the accessibility hardware and software that facilitate their use of computers in general. Sanchez and Aguayo have developed APL (Auditory

Programming Language) to enable blind children to learn how to develop programs[8], but this separate language was not intended for professional software development. Smith, Francioni and Matzek developed JavaSpeak, which assists blind students in the process of developing Java programs [11].

The original intention of the second author was to design a platform-independent programming language that was suitable for the rapid development of GUI-based applications. But the feedback from the blind programming community was unanimous. Blind programmers did not want a separate language; they wanted a scripting language that would simplify the task of designing Visual Basic forms[9].

Visual Basic Forms

Forms are the most important class of objects in a Visual Basic program. The common practice is to design the form and then write the methods that are implemented for certain events associated with the various objects embedded within the form. Thus, the inability to design a form adequately also makes it difficult to write the source code.

Visual Basic form files are stored in a text format. Although the normal way of designing forms involves “pointing and clicking”, it is possible to create the required form using a text editor and then having it added to the Visual Basic project. Even a form with only a command button requires a blind programmer to specify the height, width and position of the form’s left and top edges and the height, width and positions of the command button’s top and left edges in twips, i.e., twentieths of a point (in older versions of Visual Basic) or in pixels (in .NET versions). It is clearly difficult for anyone, sighted or blind, to create a form by writing such specifications without constantly checking to see if the form looks the way that the creator intended it to look. While this is impossible for the blind, it is also difficult and time-consuming for the visually impaired. The introduction of Visual Studio .NET in 2002[5] exacerbated the problem. Objects need entries in several places within the file and there is a procedure that cannot be modified by a text editor without compromising the integrity of the object.

DESIGN CRITERIA

Most GUI development tools such as Visual Basic and Java Swing use a similar approach: there is some kind of container class (such as VB forms or JFrames) into which other objects such as labels, text boxes and command buttons are inserted. While there are other properties that may be specified, the most important ones indicate the size and position of the object. By organizing the form into a series of rows or a series of columns, the first concern can be addressed in a way that the blind can handled easily; every object is placed sequentially in their

respective row or column. Similarly, the size options are limited to small, medium or large widths and heights specified by the number of lines of text. The positioning of the form itself is specified by dividing the screen into a 3x3 grid, using the terms left, center or right followed by the terms top, middle or bottom.

USING THE COMPILER

Form scripts all use the filename extension .fms and are compiled by typing:

```
molly ScriptName.fms
```

The form file that is generated is given the name *ScriptName.frm* (in the Visual Basic 6 version) or *ScriptName.vb* (in the Visual Basic .NET version). In both versions, the file can then be added to a Visual Basic project.

EXAMPLE OF A FORM SCRIPT

An example of a Visual Basic script is shown in Figure 1. The form's organization is rows, so each section is laid out horizontally. Although this script is written in mixed case and with indentation to clarify its structure, this is not necessary. The language is case-insensitive and while all statements are terminated by a new line (as is the case in Basic), there is no need for indentation if the programmer chooses not to use it. The form generated by the compiler is shown in Figure 2.

```
Form frmSample3
Location = Bottom Left
Caption = "Sample 3"
Organization = rows
Section
  DriveListBox Drive1
  End
  DirListBox DirLB1
  Height = 3
  Width = medium
  End
End ' Section
Section
  Frame fraTest
  Caption = "This is a test frame"
  Optionbutton optOne
  Caption = "This is opt one"
  Visible = true
  End ' Optionbutton
  Optionbutton optTwo
  Caption = "This is opt two"
  Visible = true
  End ' Optionbutton
End ' Frame
End ' Section
End ' Form
```

Figure 1 – An example of a form script used by the compiler to create a Visual Basic form

REACTION TO THE PROJECT

As stated above, the original goal of the project was altered specifically because of the feedback from the blind programming community. The prototype compiler was made available on the second author's web site (<http://home.adelphi.edu/~siegfrir/molly>) and the mailing list of blind programming.com (which is now a discussion board) was notified of its availability. It was clear from the comments that were made that the blind programmers who read the

specifications of the language and its sample scripts believed that the prototype had potential to help the blind create Visual Basic forms without the aid of a sighted person. Given that this was a prototype, there were questions about what additional features might be supported by the full working version of the compiler.

The compiler and manual are currently available online at <http://www.adelphi.edu/~siegfrir/molly>.

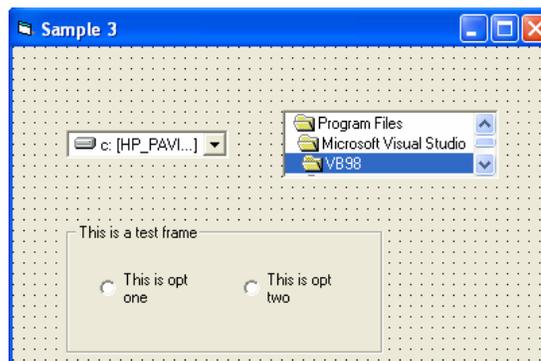


Figure 2 – The form specified by the script in Figure 1.

REFERENCES

- [1] Alexander, Steve, "Blind Programmers Face An Uncertain Future", ComputerWorld (November 6, 1998), p. 1.
- [2] Cohen, Robert F., Arthur Meacham and Joelle Skaff, "Teaching Graphs to Visually Impaired Students Using an Active Auditory Interface", Proceedings of the Thirty-Seventh Technical Symposium on Computer Science Education (Houston, TX), March 1-5, 2006, p. 279-282.
- [3] Harper, S. and S. Bechhofer, "Semantic Triage For Increased Web Accessibility", IBM Systems Journal 44(3), 2005, p. 637-648.
- [4] Hodson, Bernard, "Sixties Ushers in Program To Train Blind Programmers", ComputerWorld Canada, June 2004.
- [5] Lohr, Steve, "Microsoft Is Putting Its Muscle Behind Web Programming Tools", The New York Times, February 13, 2002, p. C1.
- [6] Riccibono, Mark A., "A Brighter Future for Blind Children", Braille Monitor, February 2004.
- [7] Rosmaita, Brian J., "Accessibility First! A New Approach to Web Design", Proceedings of the Thirty-Seventh Technical Symposium on Computer Science Education (Houston, TX), March 1-5, 2006, p. 270-274.
- [8] Sanchez, Jaime and Fernando Aguayo, "Listen what I do: Blind Learners Programming Through Audio", Memorias TISE 2004, p. 120-124.
- [9] Siegfried, Robert M., "Visual Programming and The Blind: The Challenge and The Opportunity", Proceedings of the Thirty-Seventh Technical Symposium on Computer Science Education (Houston, TX), March 1-5, 2006, p. 275-278.
- [10] Smith, Ann C., Joan M. Francioni and Sam D. Matzek, "A Java Programming Tool for Students with Visual Disabilities", Proceedings of the Fourth ACM Conference on Assistive Technologies (Arlington, VA, November 13-15), p. 142-148.