Interactive Teaching Materials in PDF using JavaScript

Petr Sojka
Masaryk University in Brno, Faculty of Informatics
Botanická 68a, 602 00 Brno, Czech Republic
sojka@fi.muni.cz

ABSTRACT

The use of JavaScript scripting language for adding interaction to portable teaching materials of a high typographical quality in PDF file format is described. An extended version of the program TEX called pdfTEX is extremely useful for such purposes. It is shown that applications similar to those done by CGI script on the web can be done in PDF, exploiting the embedded JavaScript engine implementation in PDF viewers.

Categories and Subject Descriptors

J.1 [**Administrative Data Processing**]: Education; H.5.1 [**Multimedia Information Systems**]: Animations; J.7 [**Computers in Other Systems**]: Publishing; H.5.2 [**User Interfaces**]: Interaction styles—*forms*

General Terms

Documentation, Languages, Design

Keywords

animation, interaction, teaching materials, PDF, TEX, pdfTEX, JavaScript, AcroTEX, Acrobat

1. INTRODUCTION

JavaScript is a standard way of programming web pages and customizing their behavior for a reader. Typographic quality of rendering mathematics in web browsers is still very poor. Portable Document Format (PDF) [1], on the other hand, is suitable for delivery of portable teaching materials with *programmable* behavior (animations, auto-tests) as it allows JavaScript code to be inserted into the PDF file, in addition to self-contained typographically rich content. JavaScript insertion into PDF is troublesome, and there are no development environments for interactive document programming for PDF available today.

Open source pdfTEX program [6] is very useful for this task as the TEX macro writer has commands for JavaScript code insertion at his/her disposal.

2. PROGRAMMING INTERACTIVE CONTENT FOR PDF

With techniques and tricks inspired by AcroTEX [4, 5] we have created electronic teaching materials [2] for a course on *Multivariable Calculus using Maple* for students of computer science. Several tricks were needed to prepare animations. Animations were implemented using Acrobat forms [1] support, and using document-level JavaScript routines for stepping animations forward and backward, changing the speed of animations, etc.

The full Acrobat application is needed for the development, as inserting animation images as icons into Acrobat forms cannot be done using free tools; this fact makes the debugging of such applications particularly difficult. Creation of the final PDF product cannot be fully automated without specialized Acrobat plugins.

3. CONCLUSION

Making interactive teaching materials documents in PDF using JavaScript and pdfTEX is feasible, given that for core functions, TEX macro-package is available—it hides teacher preparing animations from peculiar macroprogramming and JavaScript debugging,

Examples of animations can be downloaded from [2] as part of teaching materials (in Czech), or from http://www.fi.muni.cz/usr/sojka/animations/. For another examples of interactive content, namely the creation of online self-marking quizzes in PDF, see [3].

4. ACKNOWLEDGEMENTS

5. REFERENCES

[1] Adobe Systems Incorporated. *PDF reference: Adobe portable document format version 1.4.* Addison-Wesley, Reading, MA, USA, third edition, Dec. 2001.

[2] Z. Došlá, R. Plch, and P. Sojka. Mathematical Analysis with Maple: 2. Infinite Series (in Czech). CD-ROM, http://www.math.muni.cz/~plch/nkpm/, Dec. 2002.

[3] R. Moore. Online self-marking quizzes, pdfTEX, exerquiz. *TUGboat*, 22(3):168–179, Sept. 2001.

[4] D. P. Story. AcroTEX: Acrobat and TEX team up. *TUGboat*, 20(3):196–201, Sept. 1999.

[5] D. P. Story. Techniques of introducing document-level JavaScript into a PDF file from LATEX source. *TUGboat*, 22(3):161–167, Sept. 2001.

[6] H. T. Thành, P. Sojka, and J. Zlatuška. TEX2PDF–Acrobatics with an Alternative to DVI Format. *TUGboat*, 17(3):244–251, 1996.