High-Level Programming Tools for Interactive Mathematics

Douglas B. Meade
University of South Carolina
meade@math.sc.edu

Phillip B. Yasskin
Texas A&M University
yasskin@math.tamu.edu

Communicating Mathematics in the Digital Era
Aveiro, Portugal
15 – 18 August 2006

The Bottom Line

- Lecturers are best suited to preparing the most appropriate materials for their students.

More specifically …

The Bottom Line

- **Mathematics** lecturers are best suited to preparing **mathematically** appropriate materials for their **mathematics** students.

Outline

- Disclaimers
- Traditional Tools
- Higher-Level Tools
- Immediate Needs
- Examples
- Final Remarks

Disclaimers

- We are mathematicians
  who want to use digital media to
  communicate mathematics to the world
- We want to communicate research results to a wider audience
- We want to utilize the benefits of the digital era to improve my teaching

Traditional Tools

- Computer Algebra (CAS)
  - Excellent tools for *doing* mathematics
  - Far from optimal for *communicating* mathematics
  - Not universally available, not intuitive, not robust
- Examples
  - Maple (http://www.maplesoft.com)
  - Mathematica (http://www.wolfram.com)
  - ...

Traditional Tools

- CGI scripts and forms
  - Requires extensive knowledge of CGI / HTML / …
  - Non-trivial to connect CAS to web applications
  - License and security concerns

- Example
  - irreducibility test for lacunary polynomials
    [http://www.math.sc.edu/~filaseta/irreduc.html]

    424 lines of CGI + 327 lines of Maple + …

Traditional Tools

- Java
  - Requires programming expertise
  - Much greater control over effects and actions
  - Same concerns about CAS connectivity, license, and security

- Example
  - Tracing the locus of the vertex of a parabola
    [ParabolaVertex.html]

    225 lines of Java code

Higher-Level Tools

- Maplets
  - front-end to Java
  - still problematic to program

- Example
  - Antiderivative calculator
    [Antideriv.mw] [Antideriv.maplet]

    16 lines of Maple

Higher-Level Tools

- Embedded Components
  - more intuitive and graphical
  - weak on features
- Example
  - Irreducibility test for lacunary polynomials
    (w/ Michael Filaseta, J Algorithms, 2005; support from NSA)

    [http://maplenet.math.sc.edu/research/Irreduc.mw] [MapleNet]

    0 lines of visible Maple code

Higher-Level Tools

- Geometry Expressions
  - typical dynamic geometry interface
  - with built-in symbolics
- Example
  - Shrinking circle
    [ShrinkCircle.gx]

    0 commands (<5 minutes total time)

    *** add symbolic formula for distance
    *** copy to Maple worksheet

Immediate Needs

- User-Interface: Layout Design
  - Graphical
  - Intuitive
  - Flexible
  - Robust
  - …

Immediate Needs

- User-Interface: Functionality
  - Dynamic layout
  - Full use of traditional Java effects
    - Color and image effects
    - Default text as instruction
    - Popups
    - …

Immediate Needs

- Full integration with Internet via hyperlinks
  - to external webpages
  - to online documents
- Inter-application communication
  - grading / course management software
- Universal availability

Examples

- Maplets for Calculus
  [http://www.math.sc.edu/calclab/M4C/] [MapleNet]

  - Textbook independent (both + and -)
  - Nearly complete coverage
  - No grading capability

  - 32,327 lines of Maple programming in 70 files
    (not counting the HTML, …)

Examples

- WebALT
  http://www.webalt.net/
  http://www.webalt.com/

  - Complete online courses
  - Affordable
  - Multilingual

Examples

- Maplets for WebALT Calculus (secure)
  - Currently in pre-alpha version
  - For additional information, including access, contact WebALT or the authors

The Bottom Line

- **Mathematics** lecturers are best suited to preparing **mathematically** appropriate materials for their **mathematics** students.
  - Good mathematics requires good communication
  - Development tools must support mathematical communication
  - No intrusive overhead
  - Expectations increase as technology improves