End User Software Engineering

CHI’2007 Special Interest Group Meeting
Tue, May 2, 2007, 16:30-18:00, Room C4

Brad Myers, Margaret Burnett, Susan Wiedenbeck, Andy Ko
Definitions

• *(Note: may be controversial—defer discussion until later)*

• **Program**
  
  – “A set of statements that can be submitted as a unit to some computer system and used to direct the behavior of that system”
    
    – *Oxford Dictionary of Computing*

• **Programming**

  – “The process of transforming a mental plan of desired actions for a computer into a representation that can be understood by the computer”
    
    – *Jean-Michel Hoc and Anh Nguyen-Xuan*
Definitions, cont.

• Professional Programmer
  – Someone whose primary job function is to write or maintain software
  – Typically have significant training in programming (e.g., BS in CS)

• Novice Programmer
  – Someone who is learning to be a professional programmer
Definitions, cont.

• **End-User Programmer (EUP)**
  – People who write programs, but *not* as their primary job function
  – Instead, they must write programs in support of achieving their main goal, which is something else
  – Covers a wide range of programming expertise
    • Business executives and secretaries
    • Physicists
Examples of EUP

• Accounting (spreadsheets)
• Analysts using MatLab
• Creating a web page
• Recording Macros in Word
• Automating office tasks
• Business software (SAP programming)
• “Programming” VCRs, Microwaves
• Scientific research
• Authoring educational software
• Creating email filters
• Musicians configuring synthesizers
• Entertainment (e.g., behaviors in The Sims)
• Web 2.0: Mashups, end-user created content
Other Names

- Also called “End User Development” (EUD)
  - As in European Commission’s Network of Excellence on End-User Development
- Some “Domain-Specific Languages” (DSL)
  - Often created for end-user programmers
- Visual (Graphical) Programs
  - Sometimes created for EUP
- “Scripting” languages, “Macros”
- Rapid Application Development (RAD)
Definitions, cont.

• **End-User Software Engineering (EUSE)**
  – End-User Programming with the addition of systematic and disciplined activities that address quality issues
  • E.g., analyzing code, understanding unfamiliar code, testing code, checking code against a model, sharing code with coworkers, maintaining code, and deploying code, to name a few
End User Programmers

- A very large group

- In 2012: — Scaffidi, Shaw and Myers 2005
  - 90 million computer users at work in US
  - 55 million will use spreadsheets or databases at work (and therefore may potentially program)
  - 13 million will describe themselves as programmers
  - 3 million professional programmers

- We should make better tools for all of these people!
Evidence of Need for EUSE

• Why need Software Engineering for EUP?
• Lots of errors attributed to End-User Programming of spreadsheets:
  – Columbia Housing Authority admitted to overpaying by $118,387 due to a spreadsheet data-entry error (February 22, 2006)
  – TransAlta Corp. took $24 million charge to earnings due to cut-and-paste error in an Excel spreadsheet (June 3rd, 2003)
  – Auditor, major accounting firm:
    “...in 6 years work, checking literally hundreds of business-critical models, ... my team have never failed to find errors.”
  – .... (many more!)
  – See http://eusesconsortium.org/euperrors/
Consequences, 2

• Also, errors in:
  – Web pages
  – Email filtering rules

• From the WEUSE II workshop at CHI’2006:
  – Clinical customization package used by medical personnel reports the need for better reuse and debugging support
  – SysAdmins need better testability of database and other sorts of scripts
  – Issues with reuse of MATLAB applications
So What is Happening?

- Growing group of researchers in EUSE
- One group: The EUSES Consortium
  - End Users Shaping Effective Software
  - 7 sites, 13 researchers + ~15 students from:

  - Oregon State
  - Penn State
  - Carnegie Mellon
  - Cambridge
  - Drexel
  - IBM
  - Nebraska

http://eusesconsortium.org
Addressing the Problem

• A multi- (sub)disciplinary problem needing:
  – Software engineering research
  – Programming language research
  – Education research
  – End-user programming research
  – HCI research of all types

• This is a big job, needs a whole community!
  – Hence, this SIG, to encourage interested people to work together with us and with each other
  – Also, connect researchers and EUP vendors
Many Sessions at CHI

- This kind of work is represented at many sessions at CHI:
- Mon, 14:30-16:00, Hartmann, et. al.
- Mon, 16:30-18:00, Liu and Li
- Mon, 16:30-18:00, Hurst, et. al.
- Tue, 11:30-13:00, Session: PROGRAMMING BY PROFESSIONALS
- Wed, 11:30-13:00, Little, et. al.
- Wed, 14:30-16:00, Ballagas, et. al.
- Wed, 14:30-16:00, Session: DEVELOPMENT PROCESS
Previous Meetings on This Topic

• Feb, 2007: Dagstuhl Conference on End-User Software Engineering
• CHI 2006: 2nd Workshop on End-User Software Engineering (WEUSE-II)
• ICSE 2005: 1st Workshop on End-User Software Engineering (WEUSE)
• CHI 2005 SIG on End Users Creating Effective Software
• CHI 2004 SIG on End Users Creating Effective Software
• Many proceedings and notes available off of EUSES web page: http://eusesconsortium.org
Ways to Get Involved

• Attend upcoming events.
• Get newsletter on end-user programming/software engineering:
  – (Low-volume). Sign-up sheet going around.
• Tell us today what you’re working on.
• Help shape future events (starting with today’s discussion).
Rest of This SIG

• Summary of the discussion at the Dagstuhl Conference on End-User Software Engineering
  – Margaret Burnett

• Summary of the draft of the survey paper on EUSE
  – Andrew Ko and Susan Wiedenbeck

• What everyone is working on that is relevant to EUP and EUSE
  – YOU!
Notes from CHI’2007 SIG

• ~50 participants: The following are notes from INDUSTRIAL participants:
  • Cedric Houlett: AutoDesk – GIS product w/VB.Net
  • Yaron Periglass: CADence – for hdw/eng
  • Jeff Hoffman: Sun – tools for EUP,
    – study Fortran programmers, what HPC programmers do
    – Published in HPC forums
  • Devon Welles: Intel – “channel platform group”
    – Integration and configuration
    – Adapting products for EUP
    – System integration
  • Fred Jacobson, BMC Software – BMC Remedy Action Request System™, 4GL + VP, “no coding”, lots of screens, call it “application development”
    – Typical user would be IT person at large enterprise, or sys-admin
    – Programming or “hacking”, all levels of maturity
  • Marat Boshernitsan: Agitar Software - build unit testing tools
  • Sara Makarenko: AutoDesk – MapGuide, open source, GIS – difficult area for non-GIS experts to deal with data like this
  • Robin Jeffries: Google – Search = EUP
More Notes

• Gaby Wenneker – SAP Dev. Studio  
  – Moving more to a modelling env. for NP & P

• Anson Tsao for Microsoft VS  
  – UI Shell for VS

• Franz Schiele – SAP, business process modelling tools

• AutoDesk – Singapore, AutoCAD, vertical tools for process and power industries

• Dave Mellis: Arduino – electronics prototyping platform, for artist/designers, musical instruments  
  – Thurs, 2:30 talk

• Amy Kidd: MathWorks, user community, manage usability group for MatLab
Future Conferences

- PPIG’2007
- VL/HCC 2007
- IUI’2008
- CHI’2008