The Changing (?) Face of Programming

Brian W Kernighan
Department of Computer Science
Princeton University
"Mr. Hong describes himself as well off enough so that work is optional. He was collecting more than $1 million a year from HotOrNot, a project he and his partner had created in seven days ..."

New York Times, 10/28/07
languages
tools
interfaces
components
methodologies

"Language shapes the way we think and determines what we can think about."

Benjamin Whorf (1897-1941), *Language, Thought and Reality*
"What languages do you focus on?"

parent of a prospective undergrad
"C makes it easy to shoot yourself in the foot; C++ makes it harder, but when you do it blows your whole leg off."

Bjarne Stroustrup, creator of C++
Scripting languages

- the nature of many applications has changed
  - gluing components, web page creation

- scripting languages (Perl, Python, PHP, Ruby, ...)
  - avoid C & C++ memory management, awkward string handling, ...
  - more expressive than Java: e.g., RE's, string operators
  - casual type checking, object hierarchies, data structures

- comparatively weak tools
- slower but no one cares
  - computers are fast and cheap
  - programmers are slow and expensive
Javascript

- very weak typing
- potentially risky object model
- inconsistent implementations, poor tools
- complicated irregular interface to browser
  - no life elsewhere
- no academic credentials

But:
- very widely used in many innovative systems
  - AJAX applications, libraries, web-based programs
- illustrates important programming ideas
  - event-driven programming, callbacks, closures
  - DOM: object oriented interface with a vengeance
  - JSON: a lightweight alternative to XML for structured data
DEPARTMENT OR PROGRAM

Spring 2007 - 2008

Last Updated: 05/04/08 4:00 pm

To obtain a listing of the departments, scroll down list above and click on departments that start with:

... e.g. Select ANT-ANT present semester.

AAS - Program in African-American Studies
CLA - Classics
CLG - Classical Greek
COM - Comparative Literature
COS - Computer Science
CWR - Program in Creative Writing
CZE - Czech
DAN - Dance
EAP - Program in East Asian Studies
EAS - East Asian Studies
ECS - Program in European Cultural Studies
EEB - Ecology and Evolutionary Biology
EGR - Engineering
ELE - Electrical Engineering
ENG - English
ENW - Program in Environmental Studies
(This does not work with IE6, which appears not to understand standards. Use Firefox or Safari.)

start typing here: **cos**

(e.g., eng for English courses or qr mw for Monday/Wednesday QR courses or smith; mouse over results for details.)

<table>
<thead>
<tr>
<th>COS</th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>COS 116 ST</td>
<td>TTh 01:30-02:50</td>
<td>COMPU 105</td>
<td>The Computational Universe</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>COS 126 QR</td>
<td>TTh 01:30-02:50</td>
<td>COMPU 105</td>
<td>General Computer Science</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>COS 217 QR</td>
<td>MW 10:00-10:50</td>
<td>FRIST 302</td>
<td>Introduction to Programming Systems</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>COS 226 QR</td>
<td>MW 10:00-10:50</td>
<td>FRIST 302</td>
<td>Algorithms and Data Structures</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>COS 320</td>
<td>MWF 10:00-10:50</td>
<td>FRIEN 008</td>
<td>Compiling Techniques</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>COS 333</td>
<td>TTh 11:00-12:20</td>
<td>COMPU 105</td>
<td><strong>Advanced Programming Techniques</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>COS 398</td>
<td>-</td>
<td>Junior In</td>
<td>This is a course about the practice of programming. Programming is more than just writing code. Programmers must also assess tradeoffs, choose among design alternatives, debug and test, improve performance, and maintain software written by themselves &amp; others. At the same time, they must be concerned with compatibility, robustness, and reliability, while meeting specifications. Students will have the opportunity to develop these skills by working on their own code and in group projects. Brian W. Kernighan 41946</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>COS 423</td>
<td>MW 11:00-12:20</td>
<td>FRIEN 008</td>
<td>Theoretical Machine Learning</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>COS 424</td>
<td>TTh 11:00-12:20</td>
<td>COMPU 104</td>
<td>Advanced Topics in Computer Science: Parallel Archite</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>COS 426</td>
<td>TTh 01:30-02:50</td>
<td>EQUAA A224</td>
<td>Advanced Topics in Computer Science: Vision: From Neur</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>COS 435</td>
<td>MW 01:30-02:50</td>
<td>COMPU 105</td>
<td>Advanced Topics in Computer Science: Geometric Modelin</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>COS 444 SA</td>
<td>TTh 01:30-02:50</td>
<td>COMPU 102</td>
<td>Advanced Topics in Computer Science: Mathematical Meth</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>COS 461</td>
<td>MW 01:30-02:50</td>
<td>COMPU 105</td>
<td>Advanced Topics in Computer Science: Biological Networ</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>COS 498</td>
<td>-</td>
<td>Senior Independent Work (3.0 or 4.0 credits only)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>COS 511</td>
<td>MW 01:30-02:50</td>
<td>FRIEN 008</td>
<td>Theoretical Machine Learning</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>COS 598A</td>
<td>F 01:30-04:20</td>
<td>COMPU 402</td>
<td>Advanced Topics in Computer Science: Parallel Archite</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>COS 598B</td>
<td>M 01:30-04:20</td>
<td>COMPU 301</td>
<td>Advanced Topics in Computer Science: Vision: From Neur</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>COS 598C</td>
<td>TTh 01:30-02:50</td>
<td>COMPU 301</td>
<td>Advanced Topics in Computer Science: Geometric Modelin</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>COS 598D</td>
<td>F 01:30-04:20</td>
<td>COMPU 301</td>
<td>Advanced Topics in Computer Science: Mathematical Meth</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>COS 598F</td>
<td>F 01:30-04:20</td>
<td>COMPU 401</td>
<td>Advanced Topics in Computer Science: Biological Networ</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
"A language that doesn't affect the way you think about programming is not worth knowing."

Alan Perlis, *Epigrams on Programming*
"Give us the tools and we will finish the job."

Winston Churchill,
radio address,
February 9, 1941
Tools

- mechanization of repetitive / tedious / error-prone tasks

- **Unix introduced the conscious focus on creating and using tools**
  - small tools: grep, diff, wc, sort, uniq, ...
  - specialized tools: yacc, lex, make
  - programmable tools: shell, awk

- **Microsoft popularized IDEs**
  (integrated development environments)
  - complete systems for editing, compiling, debugging, packaging
Web Frameworks

• **software systems to create web applications**
  - client (web) front end, server code, database access, administrative subsystem, etc.

• **programs that write programs**
  - generate code from specification plus conventions

• **examples:**
  - Rails (Ruby)
  - Django (Python)
  - Google Web Toolkit (Java)
  - and zillions of others [Wikipedia lists nearly 180!!]
Google Web Toolkit

- write client (browser) code in Java
- including GUI layout, using classes like Java Swing
- test client code on server side
- once this works, compile Java to Javascript and HTML
- use the generated code on the client side
Django: Python web framework

Spare Stuff finding use for your spare goods.

Search for charities that can utilize your spare goods or excess inventory

Search bicycles, business attire, cars, laptops, music instruments, prom dresses, recliners, school supplies, tables, to

Let your spare stuff be put to good use...
Top 10 Charities

Charities are ranked by how many users click on their page for more information.

1. **Goodwill Industries Of Greater New York And Northern New Jersey, Inc.**
   Astoria, NY: Goodwill Industries is a leading provider of vocational, youth and community redevelopment services for people with disabilities and other special needs in New York and Northern New Jersey. Its mission is to expand opportunities and job capabilities for people outside the mainstream workforce, whether they are mentally or physically disabled, economically disadvantaged, recent immigrants, unskilled workers or urban youth. The agency operates its programs in the five boroughs of New York City and the metropolitan area, including Long Island and ... *(more info)*

2. **Goodwill Industries Of Southern New Jersey/Quaker City Goodwill**
   Maple Shade, NJ: Goodwill Industries of Southern New Jersey/Quaker City Goodwill is a nonprofit, community-based organization. Its mission is to provide education, job training and employment services that prepare individuals with disabilities and other disadvantaging conditions, such as welfare dependency, illiteracy, and spousal abuse, for competitive community employment. Revenue from the mass collection and resale of donated items in Goodwill's area retail stores fund employment training programs. *(more info)*
Goodwill Industries Of Greater New York And Northern New Jersey, Inc.
4-21 27Th Avenue, Astoria, NY, 11102

Goodwill Industries is a leading provider of vocational, youth and community redevelopment services for people with disabilities and other special needs in New York and Northern New Jersey. Its mission is to expand opportunities and job capabilities for people outside the mainstream workforce, whether they are mentally or physically disabled, economically disadvantaged, recent immigrants, unskilled workers or urban youth. The agency operates its programs in the five boroughs of New York City and the metropolitan area, including Long Island and Northern New Jersey. More info...

Contact Information
Email: info@goodwillny.org
Phone: 718-728-5400
Website: http://www.goodwillny.org (preferred)

Accepted Items
- clothing
- housewares
- sporting goods
- furniture
"A most important ... aspect of any tool is its influence on the habits of those who train themselves in its use."

Edsger Dijkstra, *A Discipline of Programming*, 1976
languages

tools

interfaces

components

methodologies

"Good fences make good neighbors"

Robert Frost,

Mending Wall, 1914
Interfaces

• the boundary between a service and its users

• "Every module ... is characterized by its knowledge of a design decision which it hides from all others. Its interface was chosen to reveal as little as possible about its inner workings."
  - David Parnas, "On the criteria to be used in decomposing systems into modules", CACM, 1972

• opaque types [e.g. C standard I/O library]
• objects
• web interfaces
Google Maps API

```html
<body>
  <div id="map" style="width: 800px; height: 600px"></div>
  <script type="text/javascript">//<![CDATA[
  var here = new GPoint(-76.6195160150528, 39.32743436955435);
  var map = new GMap(document.getElementById("map"));
  map.addControl(new GLargeMapControl());
  map.addControl(new GMapTypeControl());

  map.setMapType(G_HYBRID_MAP);
  map.centerAndZoom(here, 0);
  map.openInfoWindow(map.getCenterLatLng(),
      document.createTextNode("You are here, more or less"));
  //]]>
  </script>
</body>
```
You are here, more or less
The web is better when it's social

The web is more interesting when you can build apps that easily interact with your friends and colleagues. But with the trend towards more social applications also comes a growing list of site-specific APIs that developers must learn.

OpenSocial provides a common set of APIs for social applications across multiple websites. With standard JavaScript and HTML, developers can create apps that access a social network’s friends and update feeds.

Many sites, one API

Common APIs mean you have less to learn to build for multiple websites.
"Little wonder then that we see so many poorly designed APIs: it is not reasonable to expect programmers to be good at something they have never been taught."

Michi Henning, "API: Design Matters",

*ACM Queue*, May 2007
"We undoubtedly produce software by backward techniques. ... I would like to see components become a dignified branch of software engineering."

Pipes then and now

- one of earliest component implementations
  ```bash
  who | grep Joe | wc
  ```
- Yahoo pipes:
Serendipity (?)

Kernighan and Ritchie

Age 9 weeks
COM: the Component Object Model

- McIlroy's components: small libraries
- COM: objects on steroids
  - the most successful example of components in practice
Mashups: duct tape programming

- the web version of components?
- the browser as operating system?
"If I have seen further than others, it is by standing upon the shoulders of giants."

Isaac Newton, 1642-1727
"There is no single development, in either technology or management technique, which by itself promises even one order-of-magnitude improvement within a decade in productivity, in reliability, in simplicity."

Fred Brooks, "No Silver Bullet", 1986
Snake Oil?

- **structured programming**
  - top-down development, successive refinement, chief programmer teams, egoless programming
  - structured everything: design, analysis, requirements, specification, walkthroughs...

- **object-oriented programming**
  - CRC cards (Class, Responsibilities, and Collaborators)
  - object-oriented everything: design, analysis, requirements, specification, walkthroughs...

- **components**
  - RAD (rapid application development), COTS (Components off the Shelf)
  - 4th generation languages, automatic programming, programming by example, graphical programming

- **CASE tools (Computer Aided Software Engineering)**
  - UML, message sequence charts, state diagrams

- **formal methods**
  - verification, validation, correctness proofs, model checking

- **design patterns**
  - patterns of everything

- **extreme programming, refactoring, agile methods, ...**
- **aspect oriented programming**
Conclusions

• languages
  - notation matters: scripting languages are here to stay
• tools
  - mechanization matters: let the machine do the work
• interfaces
  - interfaces matter: bad APIs are easy; good APIs are hard
• components
  - reuse matters: don't reinvent the wheel
• methodologies
  - knowledge and skill matter: there is no royal road to software
The Changing Face of Programming??

Plus ça change, plus c'est la même chose