

# Reshaping the Image of Computer Science in Only Fifteen Minutes (of Class) a Week

Sara Sprenkle

WASHINGTON AND LEE  
UNIVERSITY

Shannon Duvall

ELON  
UNIVERSITY

# Lawyer Karen Sandler's heart condition means she needs a pacemaker-defibrillator to avoid sudden death, so she has one simple question: what software does it run?

Yet it turns out that it's impossible for her to see and understand the technology that's being installed into her own body and upon which her life depends. Regulatory authorities don't see or review the software either.

She simply has to trust that the vendor is telling the truth and doing things right.

In this third of four daily podcasts from [Linux.conf.au 2012 \(LCA\)](#) in Ballarat, you'll hear Sandler discuss the real-world implications of this very personal software story.

How do we know the software works as advertised? How do we know it's secure? And what will happen if something goes wrong and the vendor tries to cover up the flaws, or if the vendor just goes bankrupt and the software stops being maintained?

Sandler also discusses legal cases where the prosecution's evidence was unreliable because it relied on software that turned out to be flawed — yet another practical reason to demand the source.

You'll also hear part two of our look at [FreedomBox](#), a project to create a platform for privacy-enhancing social networks. Following on from [yesterday's episode](#), FreedomBox Foundation board member and developer Bdale Garbee gives us a status update on the project's software stack.

There's also a conversation with Mary Gardiner and Valerie Aurora about the [Ada Initiative](#), a project to increase the participation of women in open technology and culture. Also, Linux kernel developer "Rusty" Russell and geek-advocate Pia Waugh explain why the Ada Initiative and other work won Mary Gardiner this year's [Rusty Wrench](#) award for services to the Australian Linux community.



Karen Sandler  
(Credit: [Linux.conf.au](#))

<http://www.zdnet.com.au/cyborg-lawyer-demands-software-source-339330089.htm>

**Lawyer Karen Sandler's heart condition means she needs a pacemaker-defibrillator to avoid sudden death, so she has one simple question: what software does it run?**

Yet it turns out that it's impossible for her to see and understand the technology that's being installed into her own body and upon which her life depends. Regulatory authorities don't see or review the software either.

She simply has to trust that the vendor is telling the truth and doing things right.

In this third of four daily podcasts from [Linux.conf.au 2012 \(LCA\)](#) in Ballarat, you'll hear Sandler discuss the real-world implications of this very personal software story.

How do we know the software works as advertised? How do we know it's secure? And what will happen if something goes wrong and the vendor tries to cover up the flaws, or if the vendor just goes bankrupt and the software stops being maintained?

Sandler also discusses legal cases where the prosecution's evidence was unreliable because it relied on software that turned out to be flawed — yet another practical reason to demand the source.

You'll also hear part two of our look at [FreedomBox](#), a project to create a platform for privacy-enhancing social networks. Following on from [yesterday's episode](#), FreedomBox Foundation board member and developer Bdale Garbee gives us a status update on the project's software stack.

There's also a conversation with Mary Gardiner and Valerie Aurora about the [Ada Initiative](#), a project to increase the participation of women in open technology and culture. Also, Linux kernel developer "Rusty" Russell and geek-advocate Pia Waugh explain why the Ada Initiative and other work won Mary Gardiner this year's [Rusty Wrench](#) award for services to the Australian Linux community.



Karen Sandler  
*(Credit: Linux.conf.au)*

In my  

---

class, I  
could use  
this to  
discuss

---

<http://computersciencenews.blogspot.com/>



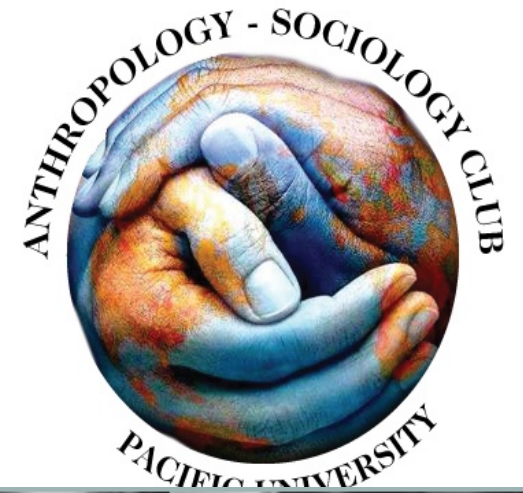
Low cost  
High  
reward



# Google Images "Computer Science"



# Google Images "Sociology"



# Goals



Put class material in real-world context

Show broad range of CS applications & opportunities

Improve written & oral communication skills

Expose how a computer scientist thinks



# Our Approach:

## Broader Issues in Computer Science

- Read recent news article before class
- Answer (written) 5 questions:
  - Interest score
  - Three important points
  - How it relates to CS in general
  - How it relates to class topics
  - Discussion question
- Discuss article in class for 15 minutes one day/week in CS0 and CS1 classes
- Pre/Post survey



# Activities & Logistics

Many options:

- Read article before class
- Have class discussion
- Blog or write about article
- Answer pre-set questions
- Present the article to others
- Find articles for extra credit

# Effectiveness Study: Results

*(Or, Why you should try this too)*

1. Students like the discussion.

*“This was my favorite part of the class!”*

In 2 different classes, a student who stopped coming to class still kept reading and blogging

# Effectiveness Study: Results

*(Or, Why you should try this too)*

2. They find computing news interesting.

*"It was interesting to see how computing really affects our lives in current times."*

# Effectiveness Study: Results

*(Or, Why you should try this too)*

3. It puts class material in real-world context.

Excerpt from DARPA Urban Challenge article:  
"We were just one IF [statement] away from success."

*"I love this part of the course. It's nice how this part allows you to see real-life examples of what you do in class."*

# Effectiveness Study: Results

*(Or, Why you should try this too)*

4. We can show our students how we think.

*"In some sense, computer science is perhaps the most pure of the sciences, simply asking, 'what is the most efficient, most precise, overall best way to solve any given problem?' regardless of what the problem actually is."*

# Effectiveness Study: Results

*(Or, Why you should try this too)*

5. It debunks student myths about CS.

*"Computer scientists aren't sitting around programming... but instead are involved in a wide variety of topics solving problems in just about every field."*

# Lessons Learned

- It really is very little effort on our part.
- Finding articles is easy & they don't have to be recent.
- You don't have to grade it.

# Lessons Learned

- Being a discussion leader can be tricky.
- Students liked larger group discussions (20-30) better than small group (3-5) discussions. (But our classes are small.)
- For small group discussions, give groups an agenda.
- Students preparing a discussion question is great backup.



# Lessons Learned

- Students will talk to others outside of class about the articles.
- *Student insights will amaze you.*

<http://computersciencenews.blogspot.com/>



Sprenkle Duvall SIGCSE12