Advances in computer mobility, connectivity and networks

October 3 2014, by Marlene Cimons

Jeannie Albrecht and two undergraduates (Danny Huang and Kelsey Levine) work on the development of Gush. Credit: Jeannie Albrecht, Computer Science Department, Williams College

Today, many people run computer applications on devices that move around, whether they are computers on buses or trains, in cars or in other transportable places, or in a pocket or purse, such as the ubiquitous Smartphone.

"We are moving to an era where computers don't have to be in a fixed location and, in fact, many are not," says Jeannie Albrecht, an associate professor of computer science at Williams College. "More and more computing devices are mobile."

But such computer mobility poses unique problems, such as intermittent network connectivity, and temporary power loss, among other things, issues that can become especially vexing when the computers are part of a distributed system, that is, a system with multiple autonomous computers that communicate through a network and interact with each other.

Albrecht, a National Science Foundation (NSF)-funded scientist, is trying to solve many of these problems, an interest that grew from her expertise in distributed computer systems. Such systems bring their own challenges, even when they aren't moving.

"If you are writing software and you want to run it on machines not physically located in one place, you have to go through many steps, including plugging in your code, installing your software, configuring the machines," she says.

While in graduate school, Albrecht built a system called Plush, for PlanetLab User Shell, an attempt to simplify some of these issues for PlanetLab, a computing "co-op" that began in 2002 at Princeton University and the University of California at Berkeley. It allowed users to perform all of these tasks, and more. It also could measure performance and react to failures.

"It almost acted like a watchdog," she explains. "It could detect when something went wrong and react to it. There was a tiny process running on each machine that would watch its operation, and talk back to a process on my machine, that I was controlling. It kept track, and it would

wake up when something went wrong on a particular machine. It could catch it and react to it."

After working with PlanetLab, she turned to another system, called DieselNet, to address the issues raised by mobility. DieselNet "literally was bunch of computers running on public city buses in Amherst, Mass.," she says, explaining that the computers were not involved in the operation of the buses, but simply installed on them and powered by the bus's battery. "They were for people who wanted to run mobile experiments, that is, experiments on machines that were not in one fixed place."

Having computers running on buses, however, meant they had power only when the buses did, and network connections varied as the buses traveled from place to place. "I wanted to extend Plush to work in an environment where you don't have constant network access, or constant power," she says. "The basic challenges are the same, but you also have these new challenges; you can't reconfigure a machine that you can't connect to."

"Can I take the same ideas—installing software, managing performance, reacting to failures—and make them work on networks whose resources are no longer in a fixed location, with constant power and continuous network activity?" she says. "We are looking at how they extend into this new context."

DieselNet no longer is available, so Albrecht has created simulations of DieselNet's operations, and is using them to conduct her experiments.

"We recorded what the buses were doing when they were moving in and out of range, and we are trying to extend Plush and develop a new platform for these mobile networks," she says.

One of her goals is to develop ways to deal with anticipated connectivity failures, both when they are predictable and when they are not, and to look at methods for coping with delays they may produce.

"Can I still manage this application, which is running on lots of mobile devices, even though some subset of the resources I am running will not be available?" she says. "We are looking at techniques for coping with the temporary unavailability of resources."

She also is studying how to incorporate advance knowledge of connectivity failures into the program.

"Say you know your application has another ten minutes to run, but you know the bus will only be running for another five. What do we do about it?" she says. "We've developed a couple of algorithms that make it easier to predict when we should abort. One of the challenges is in knowing how much time you have left to run. Some applications can pause and restart, but every application has different requirements."

She didn't begin this work with Smartphones and tablets in mind, but realizes that her research has the potential to greatly improve how they function.

"I wasn't trying to build a new app store," she says. "But if you build an app, how do you test it? The buses, in some kind of weird way, actually modeled that behavior, dealing with the challenges that [mobile devices](link) introduce. I was proposing this when Smartphones were just becoming hot. Who knew they would become as popular as they are? They really have taken off."

She is conducting this research under an NSF Faculty Early Career Development (CAREER) award, which she received in 2009 as part of NSF's American Recovery and Reinvestment Act. The award supports

junior faculty who exemplify the role of teacher-scholars through outstanding research, excellent education and the integration of education and research within the context of the mission of their organization. NSF is funding her work with $400,000 over five years.

The educational component of her grant draws undergraduates directly into the research, where they are helping her develop these new programs. Also, every other year she teaches a course in distributed systems, and introduces them to the workings of Plush, actually now known as Gush, since PlanetLab now is a part of GENI (Global Environment for Networking Innovations), an NSF program that established a vast virtual laboratory for at-scale networking experiments.

"Small schools don't always have the resources that larger schools have, but they can do all sorts of things through this one common interface, Gush," she says. "I introduce them to this tool, and show them how it works, and they get to play with some of these environments. And they love it."

Provided by National Science Foundation