

debuted; they're around the same age as Google's founders. "Today's professors grew up without search functions on their phones and computers; today's students increasingly don't remember a world without them."

"I grew up when you had to have a file; you had to save it, and to know where it was saved. There was no search function," says Saavik Ford, a professor of astronomy at the Borough of Manhattan Community College. But among her students, "they're not a generation that there's a place where files live. They just search for it and bring it up." She added, "They have a laundry basket full of laundry, and they have a robot who will fetch them ever since I got home, they went on demand." Some companies have actually played around with laundry-inclined robots, to little result.)

To a point, the new mindset may reflect a natural progression. Plavchan recalls having similar disconnects with his own professors. "When I was a student, I'm sure there was a professor that said, 'Oh my god, I don't understand how this person can have a computer under a chip on a motherboard,'" he says. "This kind of generational issue has always been around." And though directory structures exist on every computer (as well as environments like Google Drive), today's iterations of macOS and Windows do an excellent job of hiding them. (Your Steam games all live in a folder called "steamapps" — and was the last time you clicked on that?) Today's virtual world is largely a user-friendly one; people in many modern professions have little need to interact with nested hierarchies.

But in STEM fields, directory structure remains crucially important. Astronomers, for example, work with hundreds of thousands of files in the same format — which can be unwieldy to scale to a searchable system, Plavchan says.

The primary issue is that the code reviewers write, run at the command line, and have to be told exactly how to access the files it's working with — it can't search for those files on its own. Some programming languages have search functions, but they're difficult to implement and not commonly used. It's

in the programming lessons where STEM professors, across fields, are encountering problems.

Classes in high school computer science — that is, programming — are on the rise globally. But that doesn't translate to better preparation for college courses in every case. Guarin-Zapata was taught computer basics in high school — how to save, how to use file folders, how to navigate the terminal, which is knowledge none of his current students are coming in with. The high school students Garland works with largely haven't encountered directory structure unless they've taken upper-level STEM courses. Vogel recalls sovring to fit "file structure" first grade computer class, but says she was never directly taught what folders were — those sorts of lessons have taken a backseat amid a growing emphasis on "21st-century skills" in the meantime.

A cynic could blame generational incompetence. An international 2018 study that measured eighth-graders' "capacities to use information and computer technologies productively" proclaimed that just 2 percent of students had achieved the highest "digital native" tier of computer literacy. "Our students are in deep trouble," one educator wrote.

But the issue is likely not that older students are learning fewer digital skills, but rather that they're learning different ones. Guarin-Zapata, for all his knowledge of directory structure, doesn't understand Instagram nearly as well as his students do. Despite having had an account for 10 years, he's had students try to explain the app in detail, but "I still can't figure it out," he complains.

"They use a computer one way, and we use it another another way," Guarin-Zapata emphasizes. "That's where the problem is starting," Ford agrees. "These are smart kids."

she says. "They're doing astrophysics. They get stuff. But they were not getting this."

Regardless of source, the consequence is clear. STEM educators are increasingly taking on dual roles: those of instructors not only in their field, but also in basic computer fundamentals as well.

Colling's courses now include a full two-hour lecture to explain directory structure. He likens finding files to giving driving directions. He shows more than 100 students each year to pretend they're guiding others to a highlighted point. He uses every analogy he can think of.

Plavchan now also spends a lot of time teaching his students about directory structure in his courses, along with other basics, like file extensions and terminal navigation. Guarin-Zapata begins his semester with a similar tutorial. "I start with a little tour of a computer and what a computer is, what a computer is," he says. "We have memory; we have a hard drive; we have an interface; we have a file structure."

It's a difficult concept to get across, though. Directory structure is intuitive to students — it's so intuitive to professors that they have difficulty figuring out how to explain it. "Those of us who have been around a while know what a file is, but I was at a bit of a loss explaining it," said one educator in a 2019 forum post, a sentiment that respondents shared.

Ford put out a call for useful analogies on Twitter and was met with various suggestions: physical tree branches and leaves, all the books and shelves in a library. "Take their phones away and get 'em on Windows 98."

But even after presenting students with every metaphor in the books, Colling still isn't positive that his students get what he's talking about: "It

laundry basket laundry basket

PAGE: 15

feels like I'm having some success, but yeah, sometimes it's hard to tell," he says.

Plavchan agrees that there are limits to how much he can bridge the generational divide. Despite his efforts to tailor his approach, "a lot of the tools we use rely on some knowledge that our students just aren't getting."

Others, meanwhile, believe it's professors who need to adapt. Garland, who has been working with befuddled students has convinced Garland that the "laundry basket" may be a superior model. She's begun to see the limitations of directory structure in her personal life; she uses a laundry basket search function to find her schedules and documents when she's lost them in her stack of directories. "I'm like, huh ... I don't even need these subfolders," she says.

Some professors who have incorporated directory structure into their courses suspect that they may be clinging to an approach that's soon to be obsolete. Plavchan has considered offering a separate course on directory structure — but hasn't quite settled on it. "I imagine what's going to happen is our generation of students ... they're going to grow up and become professors, they're going to write their own tools, and they're going to be based on a completely different approach from what we use today."

His advice to fellow educators: Get ready. "This is not gonna go away," he says. "You're not gonna go back to the things we had. You have to accept it. The sooner that you accept that things change, the better."

laundry basket laundry basket

bridge the generational divide. bridge the generational divide.

laundry basket

bridge the generational divide. bridge the generational divide. To a point, the new mindset of student needs an adjustment to natural and expected progression.

cynic

cynic A cynic could generate a cynic and a generational divide.

another way one way

tools

change The sooner that a user accepts that things change, the better. The better the better.