

The *Communications* Web site, <http://cacm.acm.org>, features more than a dozen bloggers in the BLOG@CACM community. In each issue of *Communications*, we'll publish selected posts or excerpts.



Follow us on Twitter at <http://twitter.com/blogCACM>

DOI:10.1145/3303708

<http://cacm.acm.org/blogs/blog-cacm>

Smoothing the Path to Computing; Pondering Uses for Big Data

Members of the Computing Research Association suggest ways to broaden participation in computer science, while Saurabh Bagchi looks at use cases for big data.



Mary Hall,
Richard Ladner,
Diane Levitt,

Manuel Pérez-Quiñones Broadening Participation in Computing Is Easier Than You Think

<https://cacm.acm.org/blogs/blog-cacm/233339-broadening-participation-in-computing-is-easier-than-you-think/fulltext>
December 11, 2018

The U.S. National Science Foundation (NSF) recently introduced new requirements for the Computer and Information Science and Engineering (CISE) Directorate programs, whereby some funded projects must include a Broadening Participation in Computing (BPC) Plan. To facilitate this transition, the Computing Research Association (CRA) is launching a resource portal called BPCnet (<https://bpcnet.org>), which is being funded by NSF to connect organizations that provide BPC programs with computing departments and NSF grant proposers. These changes reflect a recognition that any significant impact

on the diversity of the field will benefit greatly from *engaging the entire academic computing research community*. Many universities will respond by expanding their broadening participation efforts to include students from groups who are underrepresented in computing, including women, underrepresented minorities, and students with disabilities (URMD). Here we list 10 small steps departments can do toward this goal.

1. Organize departmental BPC efforts at your university: Create a sign-up list of diversity activities, and incentivize faculty to participate. Create a departmental strategic plan for broadening participation that faculty can support and amplify in their funded NSF CISE proposals. Consider how to leverage BPCnet providers as part of your departmental plan.

2. Optics matter: Include pictures of URMD students in websites and printed materials. Artwork, examples in class, etc., should appeal to all students and not reinforce stereotypes. The same goes for examples you pres-

ent in class. If you think they fail to be inclusive, they probably are.

3. Make departmental infrastructure accessible, inclusive, internationalized: Provide accessible classrooms, labs, offices, websites, videos, etc. Use international alphabets for student names. Ask students for their preferred pronouns.

4. Measure and track: Analyze your enrollment, demographics, etc., regularly to identify problem areas and track changes, on your own, or with the CRA Data Buddies.

5. Create a community for URMD students: Sponsor student organizations, and send students to Grace Hopper, Tapia, and other celebrations of diversity in computing.

6. Recruit URMD teaching assistants, professors, advisors: Representation matters. Students value seeing someone who looks like them being successful in their field.

7. Promote undergraduate research: Work with women and URMD students in undergraduate research projects, such as through CRA's CREU and DREU.

8. Create curriculum enhancements that appeal to diverse students: Create introductory courses that assume no computing background, CS+X degree programs, service-learning, and accessibility electives.

9. Develop the K-12 pipeline: Work with K-12 teachers (CSTA) and improve state curricula (ECEP) to advance K-12 computing education.

10. Engage the community to stimulate computing interest and skills: Organize rigorous and joyful outreach

events that bring diverse K–12 students and their families onto your campus.

From "Increasing Diversity In Computing Is Easier Than You Think: Some Small Steps That Can Make A Big Difference," panel, 2018 CRA Conference at Snowbird, UT.



Saurabh Bagchi
Short Take: Big Data and IoT in Practice

<https://cacm.acm.org/blogs/blog-cacm/233312-short-take-big-data-and-iot-in-practice/fulltext>

December 10, 2018

Beyond the tremendous level of activity around big data (data science, machine learning, data analytics ... take your pick of terms) in research circles, I wanted to peek into some of the use cases for its adoption in the industries that deal with physical things, as opposed to digital objects, and draw some inferences about what conditions help adoption of the research we do in academic circles.

What's Driving the Convergence?

The convergence of Internet of Things (IoT) and big data is not surprising at all. Industries with lots of small assets (think pallets on a factory floor) or several large assets (think jet engines) have been putting many sensors on them. These sensors generate unending streams of data, thus satisfying two of the three V's of big data right there: velocity and volume. Next time you are on a plane and are lucky to be next to the wings, look underneath the wings and you will see an engine — if it is Rolls Royce or GE, it may even have been designed or manufactured in our backyard in Indiana. Engines like these are generating 10 GB/s of data (<http://bit.ly/2LTsMjy>) that is being fed back in real time to some onboard storage or more futuristically streamed to the vendor's private cloud. This is one piece of the IoT-big data puzzle, the data generation and transmission. This is the more mature part of the adoption story (<http://bit.ly/2SzWTz3>). The more evolving part of the big data story is the analysis of all this data to make actionable decisions, and that, too, in double-quick time.

Use Cases for Collecting Big Data

The second part of this story is in the analysis of all this data to generate actionable information. Talking to my industrial colleagues, there are five major use cases for such analysis:

1. **Predictive maintenance/down-time minimization:** Know when a component is going to fail before it fails, and swap it out or fix it.

2. **Inventory tracking/loss prevention:** Many industries of physical analog things have lots of moving parts; again, think of pallets being moved around. They want to track where a moving part is now and where all it has been.

3. **Asset utilization:** Get the right component to the right place at the right time so that it can be used more often.

4. **Energy usage optimization:** Self-explanatory, and increasingly important as the moral and dollar imperatives of reducing energy usage become more pressing.

5. **Demand forecasting/capacity planning:** Self-explanatory, but firms seem to be getting better at this at shorter time scales. Way back in 1969, the U.S. Federal Aviation Administration (FAA) was predicting air traffic demands on an annual basis (<http://bit.ly/2BWj5fv>); now think of predicting the demand for the World Cup soccer jerseys depending on which country is doing how well on a daily basis (<http://bit.ly/2R3IcHL>).

Factors Helping Adoption of Academic Research

Academia has been agog about this field of big data for, well, ... seems like forever. We academics thirst for real use cases and real data and this field exemplifies this more than most. We need to be able to demonstrate our algorithm and its instantiation in a working software system delivers value to some application domain. How do we do that? There is a lot of pavement pounding and trying to convince our industrial colleagues. Again talking to a spectrum, some factors seem to recur frequently. These are not universal across application domains, but they are not one-off, either.

1. **Horizontal and vertical.** There is a core of horizontal algorithmic rigor that cuts across the specifics of the application, but this is combined quite intricately with application-specific design choices. We can snarkily call them "hacks," but they are supremely important pieces of the puzzle. This means we cannot build the horizontal and throw it across the fence, but rather have to go the distance of understanding the application context and the vertical.

2. **Interpretability.** While ardent devotees at the altar of big data are willing to accept the output of an algorithm like the Oracle of Delphi, many of my industrial colleagues in the business of building physical objects small or large are cagey about such blind faith. Thus, our algorithms must provide some insights or knobs to play "what-if" scenarios. This sometimes runs at odds with building super-powerful models and algorithms, but it is our dictate from the real world to make smart trade-offs.

3. **Streaming data and warehouse data.** My colleagues seem to want the yin and the yang on the same platform. The data analytics routine should be capable of handling data as it streams past, as well as old data from years of operation that is sitting in a musty digital warehouse. This speaks to the need to extract value from the wealth of historical data, as well as making agile decisions on the streams of data being generated now.

4. **Unsupervised learning.** This is entering technical-jargonland, but basically this means we do *not* want to have to recruit armies of people to label data before we can let any algorithm loose on the data. That takes time, effort, legal wrangling, and we are never completely sure of the quality of labeling. So we would, whenever we can, use unsupervised learning, which does not rely on a deluge of labeled data.

Conclusion

The domains of big data and IoT are destined to mutually propel each other. The former makes the latter appear smarter, even when the IoT system is built out of lots of small, dumb devices. The latter provides the former with fruitful, challenging technical problems. Big data algorithms here have to become small, run with a small footprint, a gentle giant in the land of many, many devices.

Mary Hall is a professor in the School of Computing at the University of Utah, and a member of the Computing Research Association Board. **Richard Ladner** is professor emeritus in the Paul G. Allen School of Computer Science & Engineering at the University of Washington. **Diane Levitt** is the senior director of K–12 Education at Cornell Tech. **Manuel A. Pérez Quiñones** is associate dean of the College of Computing and Informatics at the University of North Carolina at Charlotte, and professor in the Department of Software and Information Systems. **Saurabh Bagchi** is a professor of electrical and computer engineering, and of computer science, at Purdue University, where he leads a university-wide center on resilience called CRISP.

© 2019 ACM 0001-0782/19/3 \$15.00