Satya Revisits "Pervasive Computing: Vision and Challenges"

Maria R. Ebling, *IBM T.J. Watson Research Center*
Roy Want, *Google*

In this interview with Mahadev Satyanarayanan (Satya), the founding Editor in Chief of *IEEE Pervasive Computing*, we learn about the motivation for his 2001 paper, "Pervasive Computing: Vision and Challenges" (*IEEE Personal Communications*, vol. 8, no. 4). Satya also considers here how much progress we have made and what new challenges have since emerged.

Satya has been a leader in the field of pervasive computing for many years, pioneering advances in distributed systems, mobile computing, and the Internet of Things. He is the Carnegie Group Professor of Computer Science at Carnegie Mellon University, and he is a Fellow of the ACM and IEEE. He was the founding program chair of the HotMobile series of workshops, the founding program chair of the IEEE/ACM series of conferences on Edge Computing, and the founding director of Intel Research Pittsburgh.

It has been 15 years since you published "Pervasive Computing: Vision and Challenges." Looking back, what motivated you to write the paper, and what aspects of the vision were spot on?

Certainly, the high-level prediction that this whole area of research would be very active, fruitful, and long-lived has proven true. That was the real point of the paper—to make the case that the high-level vision was sufficiently

far-out, challenging, exciting, and high-impact that excellent computer science researchers should engage deeply in it. Mark Weiser, of course, said it first and most beautifully, which is why the paper begins with a quote from his 1991 *Scientific American* paper, "The Computer for the 21st Century." With Weiser's passing in 1999, and with the Xerox PARC lab he created no longer leading the charge, I felt that his vision was unlikely to be pursued further unless the research community consciously dedicated itself to the quest.

Weiser's vision was expressed at a very high level, while my 2001 paper tried to zoom in on some key technical challenges. In addition, the decade between 1991 and 2001 had been hugely transformative in computing, with the emergence of the World Wide Web and public awareness of the Internet, the total dominance of personal computing and the disappearance of timesharing, the dire fates of flagship companies—such as the disappearance of Digital Equipment Corporation and the near-death experience of IBM—the Dot-Com boom, the end of the Cold War and its impact on DARPA's contribution to computing innovations, and so on. In technical and non-technical aspects, the computing landscape of 2001 was very different from what it was in 1991. In writing the paper, I was revisiting Weiser's high-level vision in the light of all that had happened and all that we had learned in that turbulent decade.

Looking back, it is fair to say that Weiser's vision has lived on and continues to inspire excellent research. To the extent that my 2001 paper has helped to sustain and extend Weiser's vision, it has been a success. I am grateful to have been at the right place, at the right time, to write it. The creation of *IEEE Pervasive Computing* was the most direct consequence of writing the paper, because it made clear what a rich set of open questions lay before us. That helped to bring together and create the community consensus and enthusiasm that lead to the founding editorial board. It also helped the

Published by the IEEE CS ■ 1536-1268/17/$33.00 © 2017 IEEE

IEEE Computer Society see this as a viable initiative. Looking back, it is impressive to see all the high-quality work that has been published in *IEEE Pervasive Computing* since its founding, as well as the many conferences, workshops, successful funding proposals, PhD and master's theses, and so on that have emerged in pervasive computing over the past 15 years. The number of researchers who have been actively engaged in this research is quite large, compared to where it was in 2001. Today, this content is a key part of many graduate courses worldwide.

Do you see any challenges that were not obvious at that time?

One thing that I was not thinking about in 2001 was how to sustain commercial products in this space. Today, advertising revenue is the foundation of many business models that relate to this space. Unfortunately, by its very nature, advertising is about consuming spare user attention. It therefore runs counter to the concept of lowering distraction, which was the core of Weiser's vision for ubiquitous computing. How to square this circle is a difficult challenge for which I have not seen any good solutions yet.

Were any of the challenges you saw more difficult than you originally thought?

A number of the challenges mentioned in the paper continue to be difficult. We have made progress on them but still have a long way to go, and in some cases new complications have arisen to make the problem harder. Consider battery life. In spite of advances at many levels over the last 15 years, the battery life of mobile devices is still a big concern. Experiments with Google Glass show a battery life of at most an hour when it is being put to serious use. If you use GPS on your smartphone for navigation, the battery typically lasts much less than a full day. The growing number of sensors on mobile devices, combined with many asynchronous network activities, have increased energy demand significantly. The reduction in the effectiveness of caching, mainly due to dynamic web content pertaining to advertising, and the many round trips to load a typical deeply nested webpage today, both conspire to increase network use and hence energy demand. These developments have muted the benefits of many improvements in energy demand in other aspects of mobile hardware and software. On the supply side, the energy density (joules per kilogram) of batteries is improving at a glacial pace. As early as 1997, it was noted in an NRC report that the energy density of lithium-ion batteries was half that of dynamite—the only difference is the rate of discharge. The recent anecdotes of exploding lithium-ion batteries reinforce this point. So although I did foresee battery life as a challenge, it is fair to say that the challenge has been even more difficult than anticipated.

It would not be a stretch to say that pervasive computing subsumes IoT—it covers all that IoT covers and more.

Other challenges that have proven more difficult than anticipated include measuring user distraction and inferring user intent.

In recent years, there has been great interest in the Internet of Things. How do you view the relationship between pervasive computing and IoT?

It is the same thing, with a different name and slightly different focus. The essence of the pervasive computing vision is a world saturated with sensing and computing, yet so gracefully integrated with humans that all this technology remains below their consciousness.

The essence of the IoT vision is a world full of intelligent devices that sense the physical world and integrate their observations to meet some higher level goal (such as the energy efficiency of a building or early failure warning of a jet engine). In terms of technical content relating to sensing, computing, and communication, these are virtually identical visions. The one substantial difference lies in the role of the human. Weiser placed the human at the center of his vision. Making sensing, computing, and communication technology "disappear" was his holy grail. In contrast, the IoT vision is silent about the role of humans. It would therefore not be a stretch to say that pervasive computing subsumes IoT—it covers all that IoT covers and more.

Back in 2001, cloud computing was not a popular concept. Were you surprised by the later dominance of this model, and did it address any of the challenges you presented?

Indeed, cloud computing was not on anyone's radar in 2001—except, of course, at VMware. Its emergence was unrelated to challenges in mobile and pervasive computing and was driven mainly by factors pertaining to the cost of owning and operating computing infrastructure, such as enterprise data centers. However, once cloud computing emerged, it was rapidly leveraged in mobile and pervasive computing. In fact, one of its early uses was for cyber foraging—specifically, for offloading compute-intensive operations from a mobile device. Apple's Siri speech recognition system appeared soon after the release of the iPhone in 2007. The voice signal was captured on a smartphone, sent to the cloud for processing, and the result was returned to the mobile user. Many other applications that have since appeared—including Google Voice and Google Goggles—use a similar strategy.

On the one hand, the appearance of the cloud simplified the answer to the question of where offloading should be done. At the same time, using the cloud implied large end-to-end latency. This is suboptimal for important use cases, and it led to the concept of cloudlets.

A separate use of the cloud has been the creation of "hubs" to collect data from sensors at the edge of the Internet—examples include home thermostats and water flow meters. However, cloud-based IoT hubs pose scalability challenges for high-data-rate sensors such as video cameras. They also pose privacy challenges, because users don't control the data—users are not given the opportunity to redact or denature

INTERVIEW

the data before it is released. These concerns are also addressed by the use of cloudlets.

How does your more recent cloudlet proposals to support edge computing play into the pervasive vision, then and now?

Implicit in the cyber foraging metaphor is the notion of "nearby" resources. It seemed obvious to me when writing the 2001 paper that low latency and high bandwidth to the remote execution site were essential attributes, so I never mentioned them explicitly in the paper. Proximity is thus not explicitly mentioned in the discussion of cyber foraging.

The emergence of cloud computing circa 2008 simplified some things related to cyber foraging. The view of the cloud as a single, logically centralized and managed entity that serves as a computing utility simplified questions relating to resource discovery, trust, and business models. "In the cloud" was the obvious answer to the question, "Where should remote execution be performed?"

Unfortunately, the global consolidation of cloud computing implies large average separation between a mobile device and its cloud. End-to-end communication then involves many network hops and results in high latencies. By early 2008, I realized that my implicit assumption of "nearby" in framing the cyber foraging concept was a mistake. I should have made explicit the importance of proximity.

Discussions with a number of senior researchers in mobile computing at the 2008 MobiSys conference in Breckenridge, Colorado convinced me that it was necessary to make the case for proximity explicit to the research community and to industry. In close collaboration, Victor Bahl from Microsoft, Roy Want from Intel, Ramón Cáceres from AT&T, Nigel Davies from Lancaster University, and I articulated the need for a two-level architecture for mobile-cloud convergence. The first level is today's unmodified cloud infrastructure. The second level consists of dispersed elements, with no hard state, called cloudlets. We described the cloudlet concept in the October–December 2009 issue of *IEEE Pervasive Computing* in "The Case for VM-Based Cloudlets in Mobile Computing," which has proven highly influential and has been viewed as the founding manifesto of edge computing. Cisco's more recent concept of "fog computing" is essentially the same concept.

One of the key challenges you identified focused on smart spaces. What do you think is the most successful adoption of smart spaces, and why haven't such spaces become more common?

This is an excellent question. Mobile devices are definitely much smarter today, but it is fair to say that there are fewer smart spaces than we anticipated in 2001. One reason perhaps is the economics of smart spaces. It takes substantial investment by an organization to create a smart space, such as a conference room that is richly equipped with sensors and actuators. This cost is typically borne by the organization. The level of technology is frozen for some time after the creation of a smart space. It is difficult and expensive to continuously track improvements in smart space technology. In contrast, the cost of mobile devices is typically borne by individual users. If a user carries a smartphone and wears a FitBit, then he or she pays for them both.

An excellent example of this economic tradeoff is the use of smart whiteboards. In 2001, there were many companies working on the creation of smart whiteboards that could capture what was written or drawn on them. However, the proliferation of smartphones with digital cameras has made the smart whiteboard concept obsolete. It is trivial to take a picture of a whiteboard with your smartphone before you erase it. Emailing those pictures to all participants is trivial. Optical character recognition of the images in the cloud can produce text output. A smart space component has thus been displaced by a smart mobile device.

One way to think of this is that mobile devices can be used to make a space temporarily smart. So smart spaces do exist, but they are being created transiently through the use of smart mobile devices. In general, computer vision using a combination of mobile devices and associated cloudlets can be used to make ordinary spaces "smart."

Will the implementation of physical smart spaces be entirely replaced by transient smart spaces, or is there still a place for physical smart spaces?

The ability to transform any space temporarily into a smart space is a powerful capability. A collection of colocated mobile users could bring a smart space into existence wherever they happen to be. I think we will see more ways of making ordinary spaces smart in this way. For example, at MobiSys 2014, Junjue Wang and his colleagues presented "Ubiquitous Keyboard for Small Mobile Devices: Harnessing Multipath Fading for Fine-Grained Keystroke Localization." They discussed their work on projecting a keyboard onto any flat surface for use as a normal keyboard. And at a 2015 Lenovo Techworld event, Lenovo presented prototype hardware with the ability to project directly from a smartphone onto a wall—essentially a "picoprojector" [www.youtube.com/watch?v=JwBem1Ul8dk]. The prototype can also project a keyboard onto a flat surface.

The digital camera replacing a smart whiteboard is another example. So I do believe that "on the fly" smart spaces are going to become increasingly important. Many years ago, at HotMobile 2008, Roy Want articulated the vision of "dynamic composable computing." In a way, this is a generalization of that concept to larger physical spaces.

That said, there will always be a place for physical smart spaces. For example, there might be security reasons why certain discussions should only occur within the confines of a controlled space. There might also be specialized hardware capabilities that are not portable. For example, the Icaros

harness [www.icaros.net] makes you feel like you are flying. The VR relies on a smartphone-based HUD, but the Icaros hardware provides the sensor inputs, which would need to be part of a physical smart space.

Also, large wall-sized screens, especially touch-sensitive screens, are valuable for many kinds of collaboration—the physical scale is a big part of what makes it valuable, and that is unlikely to change.

Have we succeeded in providing "minimal user distraction"?

We have not been very successful here. If anything, we have worsened the problem by using business models that rely primarily on advertising. When writing the 2001 paper, I was involved in the Aura project, which aimed to reduce user distraction. We did gain some insights and make some progress, but in hindsight, we lacked quantitative measures to guide our progress. I realize now the wisdom of the aphorism, "That which gets measured, gets improved." For a systems researcher, it is much easier to measure resources such as bandwidth, energy, and processor utilization than a fuzzy and ill-defined concept such as user distraction. So, in 2017, we are no better off than we were in 2001 about how to quantify user distraction and measure it in a working system without further distracting the user. This is an important area at the intersection of HCI and systems research.

Determining user intent is a significant challenge. How would you grade our progress toward this goal?

Our progress has been mixed. Cloud-based machine learning has been very successful in, for example, completing user queries to search engines. Being able to predict what is being searched for, without the user having to type in the entire search query, is an example of discovering user intent. It is especially valuable in mobile contexts, where a user is juggling many things and dealing with a small screen and a tiny touch keyboard. In other areas, such as using location and other sensors to predict imminent user actions and to prefetch the data needed for them, today's systems are less successful. As mentioned earlier, caching and prefetching run counter to business models that are based on forcing the user to interact with business logic in the cloud frequently (for example, to present advertisements or to track user actions at fine granularity). The obvious solution would be to place a trusted module at the endpoint that performs that business logic locally. This would reduce the frequency of cloud interactions and thus make the user experience much crisper. Ultimately, the reason to determine user intent is to improve user experience by anticipating user needs and user actions. We still have a long way to go.

The privacy risks of pervasive computing are significant, as you outlined many years ago. How much progress have we made in the intervening years? Are we building a world in which you want to live?

Sadly, I think we are in worse shape regarding privacy than we were in 2001. The practice of sending all sensor data to the cloud makes it too easy for seemingly innocuous data to escape from a user's control, before he or she realizes that sensitive inferences could be made from that data. The standard operating procedure today is to ask the user for various privacy-related permissions at the time of installing a new application. This one-time approval is too coarse grain and too far in advance of necessary context for a user to make a good decision.

Users need finer grain and more context-sensitive control of their sensor data. As pointed out recently in "Privacy Mediators: Helping IoT Cross the Chasm," an article Nigel Davies and I wrote with Nina Taft, Sarah Clinch, and Brandon Amos for HotMobile 2016, cloudlets offer a natural point of control and enforcement of privacy policies.

In general, today's business models assume that users are willing to sacrifice privacy for free services. There is rarely a way to opt-out of this tradeoff without losing the service altogether. The problem is not specific to pervasive computing; it applies across the broader computing landscape. Consider, for example, devices such as Alexa and smart toys that do speech recognition using the cloud. There is constant risk of leaking sensitive information in the use of these devices. Good solutions are needed, but I don't have any silver bullets to offer.

A possible solution path is through the decentralization and local control offered by cloudlets. As suggested in the "Privacy Mediators" article I mentioned, application-aware third-party privacy mediator software on cloudlets could inspect the raw sensor data in real time for potential privacy leaks, suppress transmission if a potential leak is detected, and log the instance for a future privacy audit of the system. This is admittedly an imperfect solution and involves many moving parts, but it is a start. I think we need to think about this problem in a way similar to dealing with viruses on desktops or laptops. Total elimination of the problem might be impossible, but the threat can be reduced to an acceptable level through vigilance and preemptive actions. P

Maria R. Ebling is a director at the IBM T.J. Watson Research Center. Contact her at ebling@us.ibm.com.

Roy Want is a research scientist at Google. Contact him at roywant@gmail.com.