Comparing autonomic and proactive computing

by R. Want
T. Pering
D. Tennenhouse

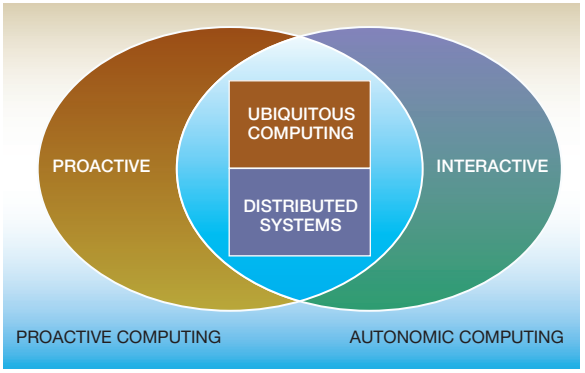
This paper provides an overview of the relationship between proactive computing and autonomic computing, considering the design of systems that are beyond the scope of our existing computational infrastructure. Autonomic computing, as described by IBM's manifesto on the subject, is a clear statement of the difficulties and challenges facing the computing industry today. In particular, autonomic computing addresses the problem of managing complexity. Intel Research is exploring computing futures that overlap autonomic computing but also explore new application domains that require principles we call proactive computing, enabling the transition from today's interactive systems to proactive environments that anticipate our needs and act on our behalf.

Autonomic[1] and proactive[2] computing both provide solutions to issues that limit the growth of today's computing systems. In the 1990s, the ubiquitous computing vision[3] extended what has been traditionally called distributed systems, a field in which the application focus has been primarily office automation.

To date, the natural growth path for systems has been in supporting technologies such as data storage density, processing capability, and per-user network bandwidth, with growth increasing annually for 20 years by roughly a factor of 2 (disk capacity), 1.6 (Moore's Law), and 1.3 (personal networking; modem to DSL [Digital Subscriber Line]), respectively. The usefulness of Internet and intranet networks has fueled the growth of computing applications and in turn the complexity of their administration. The IBM autonomic vision seeks to solve some of the problems from this complexity by using eight principles of system design to overcome current limitations. These principles include the ability of systems to self-monitor, self-heal, self-configure, and improve their performance. Furthermore, systems should be aware of their environment, defend against attack, communicate with use of open standards, and anticipate user actions. The design principles can be applied both to individual components and to systems as a whole, the latter providing a holistic benefit that satisfies a larger number of users.

At Intel Research we enthusiastically support the aims of autonomic systems and at the same time consider how computing systems will be used in the future. To date, the familiar personal computer (PC) infrastructure has been applied most effectively in the realm of the office and the home. Going forward, we are intrigued by other areas of human endeavor that are ripe for the application of computer-based technology. Proactive computing extends our horizon by recognizing a need to monitor and shape the physical world, targeting professions that have complex real-world interactions but are currently limited by the degree of human involvement required. We are addressing some of the challenges that exist beyond the scope of earlier ubiquitous computer systems to enable future environments involving thou-

Figure 1 The relationship of computing paradigms



sands of networked computers per person. Proactive system design is guided by seven underlying principles: connecting with the physical world, deep networking, macro-processing, dealing with uncertainty, anticipation, closing the control loop, and making systems personal.

An emphasis on human-supervised systems, rather than human-controlled or completely automatic systems, is an overarching theme within proactive computing. Computer-to-user ratios have been changing over time: 1:many turned into 1:1 with the advent of the PC in the 1980s, and into many:1 with the explosion of mobile devices in the new millennium. Currently, most people in the United States typically own (sometimes indirectly) many tens of computers, ranging from portable devices to consumer electronics. These systems compete for human attention, an increasingly scarce resource in modern living. Before the sheer number of devices overwhelms us, solutions need to be found to remove people from the control loop wherever possible, elevating their interaction to a supervisory role. One way would be to use pure artificial intelligence, a lofty goal that will not be attainable in the near future. Proactive computing, therefore, focuses on human-supervised operation, where the user stays out of the loop as much as possible until required to provide guidance in critical decisions.

A simple present-day example that illustrates a human-supervised system is a modern home central heating system. Such systems typically have a simple regime for morning, day, evening, and night temperature settings. Normally, the system operates untended and unnoticed; however, users can readily override these settings at any time if they feel hot or cold, or to address an impending energy crisis. Furthermore, if the system were instrumented with a sensor network and knowledge of a family's calendar, the temperature and energy consumption could be optimized proactively to allow for in-house microclimates, late workdays, and family vacations. However, extending this example to more complex systems is quite a challenge—most decisions do not simply become a selection between "too hot" or "too cold."

As illustrated in Figure 1, there is considerable intellectual overlap between research into autonomic and proactive systems. Both autonomic and proactive systems are necessary to provide us with tools to advance the design of computing systems in a wide range of new fields. In the following sections we provide an overview of the issues, technology directions, and examples of why both these visions are necessary.

Extending the application domain

Enabling a computing future that goes beyond the current in-home and in-office application domains will require the adoption of new design principles. Here we examine three of the seven proactive principles: connecting with the physical world, real-time/closed-loop operation, and techniques that allow computers to anticipate user needs. Readers interested in the remaining four are directed to a description of proactive computing on the Web at http://www.intel.com/research.

Connecting to the physical world. Most of our computing infrastructure to date connects personal computers through networks to arrays of servers. The resulting systems provide us with a virtual environment allowing us to author, process, and file information, which, through people, can have an indirect influence on the physical world. To enable a world in which computing aids us in our day-to-day tasks, the physical world must be instrumented so that computer systems can have direct and intimate knowledge of our environment, ultimately using that information to effect change. Corresponding examples are microclimate weather forecasts, monitoring road traffic, and determining where people might be located in an earthquake-damaged building.

Needless to say, there are a number of inherent problems in building such a system. First, there are pragmatic issues such as maintenance, connectivity, and

finding suitable power supplies. Second, we are describing systems that, when applied on a city or national scale, have never before been built. The coordination and management issues take on a new level of difficulty; new protocols need to be created to enable appropriate data flow; and power management becomes a critical parameter for sensors that must operate from independent energy sources. Applying sensors to the physical world on a national or global scale is a daunting task, but as our societies become more complex and population densities increase, the payback will be worth it.

Scaling systems to a size large enough to monitor the physical world raises immediate problems of administration and utilization—the very problem that autonomic computing sets out to solve—and we cannot simply look to existing computer systems for guidance. However, by using simple nodes that can be individually and comprehensively characterized, it may be possible to learn more about the techniques required to maintain larger networks of conventional computers, informing both proactive and autonomic system builders. Multihop wireless sensor networks, such as the networks we are working on in collaboration with our colleagues at the University of California, Berkeley,[4,5] have exactly this characteristic.

Real-time and closed loop operation. If we expect our computers to become more integrated with the physical world, real-time response will become a critical factor that needs to be supported by all computer systems. In the 1960s, computer systems were either fully interactive, putting humans in the control loop, or completely inflexible, built on a dedicated control system. In order to integrate systems fully into real-world tasks, the systems must be able to respond faster than is possible with a person in the control loop: they must have real-time response to physical-world events.

If general-purpose computing systems were redesigned to make real-time guarantees, many new proactive applications would be possible and perhaps even begin to appear as mass-market shrink-wrapped software in major retail stores. However, the underlying issue is that most software systems make no guarantee of a real-time response, instead hiding behind layers of abstraction without considering the response time induced by varying conditions. Those of us familiar with the embedded systems world typically resort to specialized software based on real-time operating systems (RTOS) for critical control applications, capabilities that are not supported by most general platforms.

Anticipation. Anticipation is a cornerstone of proactive computing. For systems to be truly proactive, they need to in some sense predict the future. Our research is currently focusing on the use of context, statistical reasoning, and data-handling, all summarized below, as a baseline for anticipating a user's needs. Utilizing these techniques, and others, will allow systems to quickly handle real-world situations and provide the appropriate level of user interaction.

Context aware operation. Portable and wirelessly connected systems have opened up the opportunity to use contextual information, such as physical location and the availability of surrounding infrastructure, to modify the behavior of applications. Both autonomic and proactive systems can take advantage of context by using the environment in which they operate to guide policy decisions. Autonomic computing can be of benefit directly in supporting new configurations, for example, through the local discovery of resources and setting up default operation. Proactive systems, working at a higher level, can filter information for display and customize the effects of commands.

Location is one of the most useful parameters to define context, and making high-fidelity location information available to mobile devices and their supporting systems is one of our immediate research goals. Some of our research programs are looking at ways to track the location of objects inside a building (beyond the capabilities of GPS, the Global Positioning System), taking advantage of the properties of existing wireless networks, or finding solutions for augmenting environments in a cost-effective way. We are also developing a location representation and application interface that allows common access to the data, essentially examining the type of protocol stack that might be useful as a standard to fuse and disseminate location information.

Statistical reasoning. In the last decade there have been advances in analytical techniques that use statistical methods to solve important problems. These techniques have expanded and even replaced some of the more traditional approaches using deterministic methods. Examples of applied techniques include Hidden Markov Models, genetic algorithms, and Bayesian techniques. We believe there is considerable benefit in applying these techniques to the management and analysis of large systems, both in

the information technology field and for process control and manufacturing in industry.

In some World Wide Web applications such as the Google search engine, these techniques are already being applied to data mining. Other successful areas of computer science that use statistical techniques are speech recognition, vision processing, and even the routing algorithms used by some computer-aided design tools. Moving forward, we will apply statistics to information contained in the physical world on a real-time basis.

Proactive data-handling. The exponential increase in the density of data-storage technology and the increasing network bandwidth available for data transport provide the means for proactive computing systems to quickly provide data to users without their explicit intervention. Proactive computing systems can take advantage of high-density portable storage that allows systems to prefetch data, which might be useful to users in the future without burdening them with a cumbersome mobile device. Likewise, high-bandwidth networks can move bulk data to a server physically near a user in a short period of time—a technique we call data staging. However, autonomic techniques must make certain that users are able to trust such systems by ensuring they are able to operate under a wide variety of conditions.

Both local data caching and data staging can play a vital role in supporting user mobility. Networks provide invaluable up-to-date connectivity, but if relied on completely, will sometimes fail a user when they are not available or become congested. Local data caching, in contrast, can serve a user if the cache contents are well chosen, but the data may not always be the latest version. By utilizing both of these techniques, proactive computing aims to make data available to computers moving through the physical world in real time, thus supporting the overall vision.

Catalyzing research

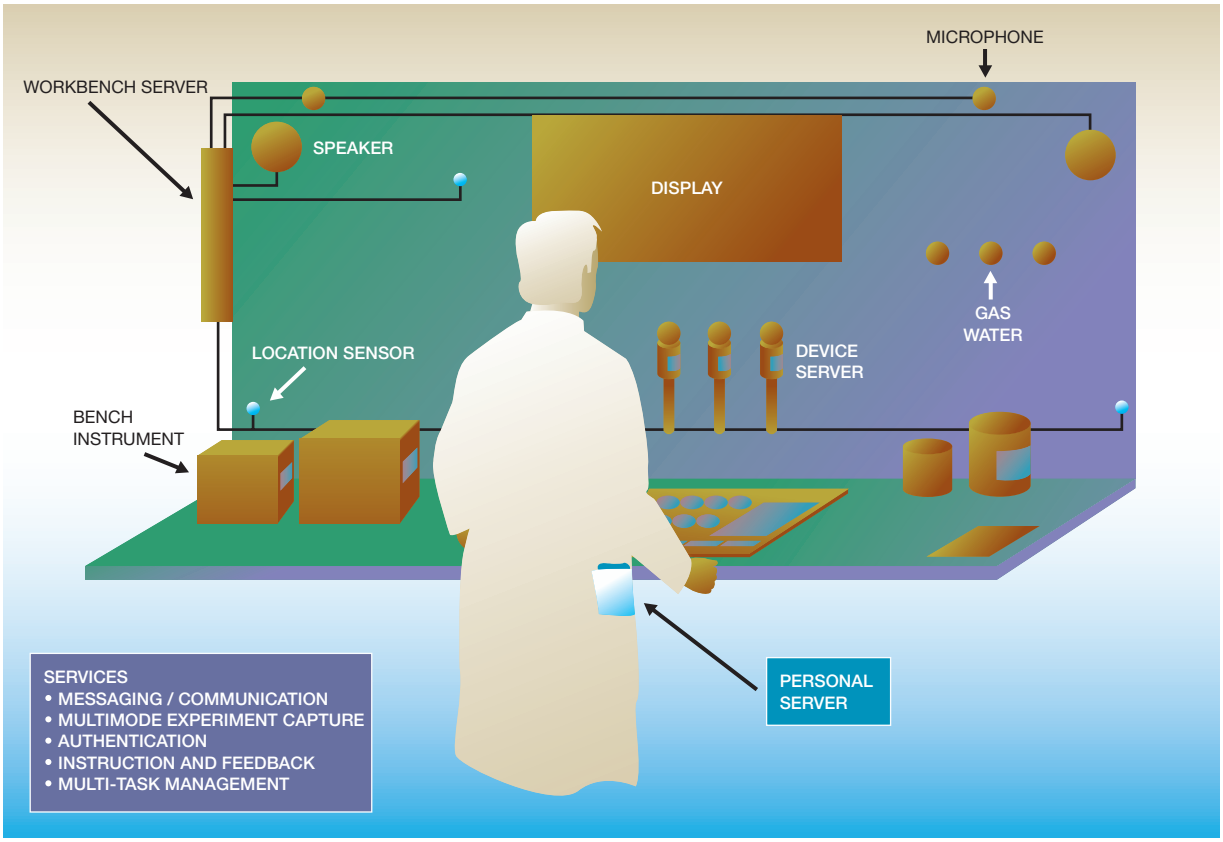
In order to embrace these challenges, we briefly summarize two of the projects that are part of Intel Research's project portfolio and designed to drive research and the use of computers beyond traditional environments.

Labscape. This project (in collaboration with the University of Washington) sets out to augment a microbiology laboratory and automate the recording and analysis of results, a prime example of an environment for which computing has had little impact. Labscape[6] sets out to instrument reagents, reaction vessels, test equipment, and the staff, and track their relative location during the experimental process (Figure 2). During any experimental procedure, many processes need to be recorded in a laboratory notebook; however, some steps are sometimes omitted, and sometimes, cross contamination occurs between reagents. In a traditional laboratory these failures can only be tracked down by skilled staff; there are no inherent mechanisms for monitoring status. With Labscape, the whole experiment can be recorded electronically and automatically generate a notebook entry for the method and results. The benefit is that no steps are accidentally lost, and furthermore, an expert system can examine the data for potential contamination risks and other experimental pitfalls.

In Labscape, a complex web of computation is created as the result of many communicating components. The principles of autonomic computing are essential as the underpinning for these systems, enabling the components to reliably and efficiently cooperate with one another. However, it is also beyond the scope of traditional computing environments touching the physical world, needing real-time response, and keeping the user out of the computational loop wherever possible. Thus proactive computing plays a vital role in the management and coordination of such a system, making inferences and using context to record data and assess risk.

The Personal Server. The Personal Server[7] focuses on a user's interaction with personal mobile data through the world around him or her, inspired by the trends in computation, storage, and short-range wireless communication standards. The underlying thesis is that storage density, which is doubling annually, will lead to one-inch disks that may store over one terabyte by 2012. With this information density available, it will be possible to carry vast amounts of data in one's pocket, some of which may really be needed, and other information that might be included, just in case. The device, which we call a Personal Server (Figure 3), can be small enough that a person will always have it available, perhaps embedded in a cellular phone or worn as jewelry. Because it does not rely on an integrated display as its primary interface, it can be quite small and still provide rich interaction. It is designed to take advantage of the surrounding computing and display infrastructure, allowing information to be opportunistically viewed on neighboring displays, thus free-

Figure 2 Labscape—instrumenting a microbiology laboratory (used with permission of Larry Arnstein, University of Washington)



ing users from carrying the bulk and weight of a screen. Standard wireless protocols, which typically provide the mechanism for sending data between the device and the host, can be used in an *ad hoc* and proactive way to discover useful information in the environment and record it for future use. Similar opportunities occur when a Personal Server encounters other Personal Servers that may advertise particular information, enabling personal peer-to-peer sharing.

Once again autonomic principles are required for this system to be successful, establishing a sense of self for the Personal Server and guarding against possible adverse data or programs that may be pushed onto it. In addition, proactive techniques are required for predicting the types of data the user needs, based on previous data access patterns or the context of the user.

Figure 3 A Personal Server prototype

The future of computing

Computing has reached a point where conventional office-bound information technology is no longer the main driver for the expansion of computational infrastructure. There are many tasks, both exotic and mundane, that can benefit from applied computation. The networking of embedded computers will unlock data that are presently stranded and allow us to apply computation beyond traditional boundaries. As these data flow into larger systems, new opportunities will be found to bring about productivity gains from the data and to offer new services that impact our lives. However, dealing with thousands of processors per person, and the torrents of data that they provide, will force us to move from interactive to proactive paradigms. This move is the aim of the proactive computing program at Intel Research, which encompasses activities in universities and industry alike to develop mechanisms that support proactive behavior.

It is clear that many of the examples we have described will also rely on the principles of autonomic computing, because they have an inherent need for self-configuration, self-healing, and self-monitoring. These factors are necessary for scalable systems and thus are integral to both endeavors.

We have all enjoyed an exciting ride as the computing industry has moved faster than any other in history in terms of its technological progress, mainly because of the exponential growth factors driving higher processor performance, increasing memory density, and lowering power consumption. However, the ride is far from over, and under the auspices of autonomic and proactive techniques, computers will take us in some very new directions.

Acknowledgments

We would like to acknowledge all members of Intel Research and our colleagues at IBM whose work focuses on turning the visions of autonomic and proactive computing into reality.

Cited references

1. P. Horn, *Autonomic Computing*: *IBM's Perspective on the State of Information Technology*, IBM Corporation (October 15, 2001); available at http://www.research.ibm.com/autonomic/manifesto/autonomic_computing.pdf.
2. D. L. Tennenhouse, "Proactive Computing," *Communications of the ACM* **43**, No. 5, 43–50 (May 2000).
3. M. Weiser, "The Computer of the 21st Century," *Scientific American* **265**, No. 3, 94–104 (September 1991).
4. S. Conner, L. Krishnamurthy, and R. Want, "Making Everyday Life Easier Using Dense Sensor Networks," *Proceedings of Ubicomp 2001: 3rd International Conference on Ubiquitous Computing*, Atlanta, GA, Springer, Lecture Notes in Computer Science **2201** (October 2001), pp. 49–55.
5. D. Estrin, D. Culler, K. Pister, and G. Sukhatme, "Connecting the Physical World with Pervasive Networks," *IEEE Pervasive Computing* **1**, No. 1, 59–69 (January–March 2002).
6. L. Arnstein, C. Hung, R. Franza, Q. Zhou, A. LaMarca, G. Borriello, S. Consolvo, and J. Su, "Labscape: A Smart Environment for the Cell Biology Laboratory," *IEEE Pervasive Computing* **1**, No. 3, 13–21 (July–September 2002).
7. R. Want, T. Pering, G. Danneels, M. Kumar, M. Sundar, and J. Light, "The Personal Server: Changing the Way We Think about Ubiquitous Computing," *Proceedings of Ubicomp 2002: 4th International Conference on Ubiquitous Computing*, Goteborg, Sweden (September 30–October 2, 2002), Springer, Lecture Notes in Computer Science **2498**, pp. 194–209.

Accepted for publication October 11, 2002.

Roy Want *Intel Corporation, 2200 Mission College Boulevard, Santa Clara, California 95054 (electronic mail: roy.want@intel.com).* Dr. Want is a principal engineer at Intel Research. His interests include ubiquitous computing, wireless protocols, hardware design, embedded systems, distributed systems, automatic identification, and micro-electromechanical systems (MEMS). He received his B.A. degree in computer science from Churchill College, Cambridge University, UK, in 1983 and continued research at Cambridge into reliable distributed multimedia systems. He earned a Ph.D. degree in 1988 and then went to Olivetti Research (1988–1991). While there, he developed the Active Badge, a system for automatically locating people in a building. Dr. Want joined the Ubiquitous Computing program at Xerox PARC in 1991 and led a project called PARCTab, one of the first context-aware computer systems. At PARC, he managed the Embedded Systems group and earned the position of principal scientist.

Trevor Pering *Intel Corporation, 2200 Mission College Boulevard, Santa Clara, California 95054 (electronic mail: trevor.pering@intel.com).* Dr. Pering is a research scientist at Intel Research. His research interests include many aspects of mobile and ubiquitous computing, including usage models, power management, novel form factors, and software infrastructure. He received his Ph.D. in electrical engineering from the University of California, Berkeley, with a focus on operating system power management techniques. Outside of engineering, he enjoys music (he plays jazz trombone), backpacking, and travel. He is a member of the ACM.

David Tennenhouse *Intel Corporation, 2200 Mission College Boulevard, Santa Clara, California 95054 (electronic mail: david.tennenhouse@intel.com).* Dr. Tennenhouse is an Intel Vice President and Director of Research. He has been one of the pioneers of asynchronous transfer mode (ATM) networking, active networks, software radio, and desktop media processing. He previously served as Chief Scientist and Director of the Information Technology Office of the Defense Advanced Research Projects Agency (DARPA), where he formulated the PRO-Active Computing research strategy of DARPA, which emphasizes the networking of embedded and autonomous systems. Dr. Tennenhouse received his B.A.Sc. and M.A.Sc. degrees from the University of Toronto. In 1989, he completed his Ph.D. at the Com-

puter Laboratory of the University of Cambridge, UK. He then joined the Massachusetts Institute of Technology, where he held appointments in the Department of Electrical Engineering and Computer Science and in the Sloan School of Management. He is a founder of a consulting firm with expertise in fault tolerant transaction processing and has been a consultant to a number of organizations. Dr. Tennenhouse is a member of the ACM and IEEE and served on the Visiting Committee on Advanced Technology of the National Institute of Standards and Technology. He has been a member of the Subcommittee on Computing Information and Communications R&D of the National Science and Technology Council and chaired the Technology and Policy Working Group of the President's Information Infrastructure Task Force.