Emulating the Future with/of Pervasive Computing Research and Development

Abstract

This position paper presents a vision of Pervasive Computing as a complex and user-centric research and development object. Emphasis is put on an "augmented emulation" as a core toolkit for development and assessment of future work and we will place our proposal in the frame of some tools and work that we have undergone. We believe that developing a use-case/scenario methodology on top of this toolkit will allow evaluation of the developed technologies by using a combination of real products, prototype softwares, emulated softwares, hardware and environment in interaction with real users or agents. Those scenarios are ways of describing application needs. The refinement of elements of those large-scale real-time augmented simulations should rely on standard metric (domain specific) and user-related metrics, and will lead to more refined scenarios in a global process.

Keywords

Pervasive/Ubiquitous Computing, Ambient Intelligence, large scale emulation, augmented simulation, model, standard, taskforce

Author

Laurent Ciarletta (laurent.ciarletta@loria.fr)
+33 -3-83-59-20-11
Madynes Research Team,
LORIA,
Technopole de Nancy-Brabois - Campus scientifique
615, rue de Jardin Botanique - B.P. 101
54600 Villers-Les-Nancy Cedex
France

1. Introduction

Pervasive or Ubiquitous Computing can be seen as the point of convergence of four classical Computer Science areas (cf. figure 1): Networking (connecting the elements, accessing data), Embedded Computing (constantly improving software and hardware miniaturization and autonomy), Personal Computing (providing services) and Computer-Human Interaction (with Artificial Intelligence providing the needed context-awareness and automatic customization).

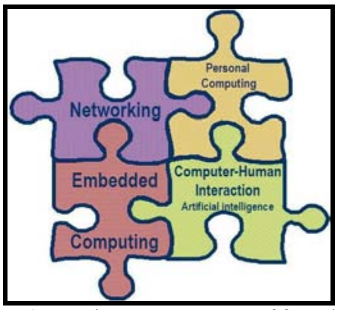

Figure 1: Pervasive Computing = convergence of four classical Computer Science areas

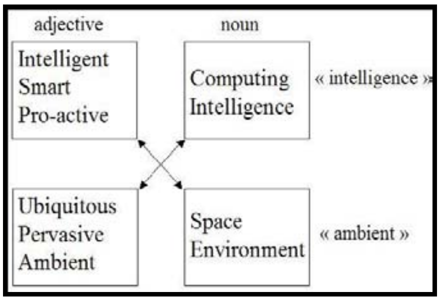

Figure 2: Pervasive Computing, Smart Space, Pro-active Computing, Ambient Intelligence etc.

Pervasive Computing is the idea that computing power ("intelligence") and connectivity will be available everywhere and at anytime ("ambient") and help users in a transparent way in their everyday-life. The association of these two concepts can be seen in the (too?) many expressions that are used to designate it (cf. figure 2). This prospect that computers will be everywhere and give the environment a lot more (computing power) intelligence and interactivity with human users has generated a large amount of interest in the last years as a source of inspiration in Computer Science research and can be considered as the Eldorado where researchers can experiment their wildest ideas.
The paper is organized as follow: limits of real test-beds are evocated to motivate the interest of emulation as a key underlying engine and fuel for Pervasive Computing research. Next, we analyze existing usage scenarios of Pervasive Computing and emphasize the need for making this technology tangible for everyone. Then we suggest building on the classical models and standards to propose a framework that will bring together the pieces of Pervasive Computing and its users to better evaluate pervasive applications.

2. Emulation augments reality

The Limits of Real Test-beds

Research projects, both academics and industrials, have built real test-beds (especially in the US) from a room scale (SmartSpace at NIST [1], Microsoft Easy Living [2]) up to a residential condominium (MIT Changing Places/House_n project [3] or Fraunhofer inHaus [4]). While such environments offer impressive demonstration capabilities, they present important shortcomings:

- Obviously, they are expensive and difficult to set up (hardware cost, need for a handyman…), an investment that most research teams can't afford.
- As a consequence of the previous point, the scale of real test-beds is limited.
- They fall obsolete quickly (sometimes even before they are finished).
- They can not be (easily) moved from one place to another. That means that only the "locals" can experiment on it. It minimizes the chances for collaboration with other research teams around the world.
- It is difficult to replay the exact same scenario over and over into them because many parameters vary (the speed at which a person moves into the corridor, the day light, changes in the test-bed …).
- They are usually tied to a particular usage and it's arduous to adapt them for another (for instance: a smart kitchen environment can't be converted into a smart class room overnight!).
- They are not well-suited for specific applications such as those related to handicapped, seniors or children mainly for safety reason.

Generalized Emulation

Definitions of "to emulate":

1. Effort or ambition to equal or surpass another
2. To strive to equal/match or excel, especially through imitation
3. (Computer Science) To imitate the function of another system (not necessary at the same speed)

Firstly, the fact that various communities are involved is an opportunity to "emulate" each other, to blend the borders that have been built and to work on common objectives and grounds. There is here a need for common vocabulary,

testing technologies, standard interfacing, interoperability and interaction solutions. Those are the foundations for an open, strong and dynamic Pervasive Computing community.
Secondly, since Pervasive Computing applies to complex systems (from low-level technologies to user interactions), in order to specify good applications, it would be interesting to completely emulate those systems creating fake worlds where the specific piece being developed can be embedded, tested, compared with other solutions and demonstrated in its context, even though some of the technologies have not been developed yet, or are available as prototypes on a small scale. We are voluntarily using generalized emulation instead of augmented (with real world measures) simulations since we have the prospect of gradually interacting with the real-world and real-users in real-time.
For instance, imagine you develop an innovative service, such as an urban emergency system. This system needs pervasive high-speed network connectivity and context-aware support, since its goal is to inform the closest medical and police forces available in the case of an accident. A complete scenario with an emulated environment (physical layout and properties of objects), and users (using intelligent agents for example) as well as devices and sensors emulators would obviously allow focusing on the development of the application and related technologies, and to experiment several solutions. But perhaps more importantly, it would allow others to provide and test the missing bricks (for example future devices, sensors, expert systems), the future environment (for example an architectural project for a given neighborhood) and to demonstrate "how it would and should work" (as far as needed/critical infrastructures, technologies and performances are concerned).
Our inspiration comes from looking at architecture, where entire neighborhoods are simulated, from the mechanical point of view to the environmental impact, or engineering, where behavior of cars and ergonomic are simulated, or from some fields of Computer Science like networking. How can we bridge all of these?
Some of the bricks are already available today, like:

- hardware emulators for PC-like devices (PocketPC/Palm emulators [5], VMware [6], User Mode Linux [7], Bochs [8], to name a few),
- electronic circuitry design toolkit,
- network simulators/emulators (NS2 [9], NCTuns [10], NISTnet [11], etc.),
- low-level wireless signal propagation simulators (WiSe [12]),
- Virtual Reality toolkits [13] and game engines for simulating the environment,
- Tangible User Interface [14] toolkit for prototyping physical interactivity,
- Intelligent agents [15] that can be used to emulate users in well defined constrained contexts.

What's missing is some common platform where different emulators, real hardware and applications would work together to build complex simulations. Depending on the scale of the distributed nodes and of their performances, bridging the different emulators can go from loosely coupled to constrained real-time architectures.
The need for a common platform starts with some development and testing tools and will naturally lead to (and nourish at one time) the question of modeling, measurement (metrics) and standardization.

3. Putting the user in the center

It's particularly important for Pervasive Computing to evaluate the perception its users have of it. Looking for "successful" applications and products of Pervasive Computing, and surveying which ones are making titles in magazines and TV shows or are really being commercially offered, one founds: smart fridges [16], connected multicolor "mood" lamps [17], PDAs and smartphones, robotic mowers or vacuum cleaners [18], robotic pets and friends, PAC-MAN in real city [19], simple location-aware wireless phone services and recently RFIDs. Lots of them look like gadgets for nerds, or really expensive toys. So a basic analysis of these applications may show that killer application is all about entertainment, and is it difficult to see the impact on human well-being.
Surely, Pervasive Computing is mainly confined to entertainment because remaining issues (such as security) prevents it from entering a critical field (such as medical), but also maybe because it can not be convincingly demonstrated to "serious users": one of the pitfalls of promising technologies is that the first targeted/potential users are actual researchers expressing what they would like to see in the future. To other people, this often looks a lot like a Sci-Fi writer having the opportunity to implement his own ideas, but it's not tangible.
A solid set of modeling and experimentation tools can create a situation where this creativity can express its potential, and where other users (researchers, customers, or sponsors) may experiment with it progressively and bring their input in the process. It can be used at the same time as an educational and demonstration (advertisement) tool. This could also provide understanding and therefore confidence in the technology. Furthermore, all the classical well known objective metrics continue to be available with such tools, and more subjective metrics such as usability and quality feedback based on users "feelings" could be added.

4. A Generic High Level Model for the Pervasive Puzzle

When browsing Pervasive Computing research, subjects range from RFID technology to Graphical Interface Ergonomic passing by agent technologies. It may be difficult at times to see how all of these fit into the global scheme.

Following a natural tendency, almost every field of Computer Science, every community develops its own vision of Ubiquitous Computing, and is somehow envisioning further research through the prism of its domain.
A generic high level model could be used and extended to provide a better understanding of the relative positioning of the domains and where their interactions take place. Tools similar to the HLA (High Level Architecture [20]) used in military simulations, or its lighter versions like the MSI (Multi Simulation Interface [21] could provide this federative capabilities and could help organize distributed emulation by logically placing elements of the simulations relatively to each other.

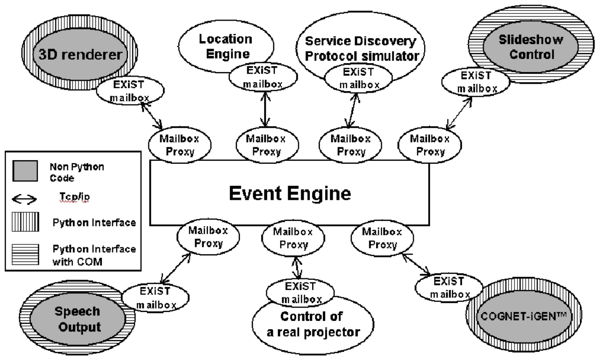

Figure 3: EXiST in action, simulating a smart conference room

In [22] we propose a proof of concept of such a generic toolkit: to demonstrate a smart conference room, EXiST (EXperimental Simulation Tool) combines a real wireless projector (Aroma smart prototype projector using Jini software [23]), intelligent agents to represent users, OpenGL modules (using Blender) to describe and visualize the physical layout of the room and other modules as depicted in figure 3. EXiST is conceptually tied to the LPC (Layered Pervasive Computing) model [24] which generalizes layered models such as the ISO-OSI for Networking taking into account the environment at the bottom part as well as the user goals and application design purpose in the highest layer, but leaving the lower granularity to specialized fields, where specific models remains more pertinent.

5. Synthesis and propositions

The variety of Computer Science fields and technologies involved in Pervasive Computing makes its richness and complexity. Lots of researches are conducted in parallel, and the results of one may be needed by others, which often limits the field of possible applications and the speed of iterations. When considering that Pervasive Computing is User-centric, new evaluation techniques (combining exisiting ones, using emulation) and metrics (such as usability) have to be considered. But researchers should not see themselves as standard users since it can hinder a broader impact of their research seing as hype (or worst). Applications are the visible tip of the iceberg, and the challenges they offer can uncover main issues and lead the underlying tehchnologies development.
We propose to follow two parallel paths to allow the emergence of visible (demonstrable) and evolved application-oriented research and development:

- use large scale (both vertically and horizontally[1]) emulations/simulations based on scenarios to test, demonstrate and emulate creativity for applications. A complete toolkit can be defined reusing lots of already existing tools. The bridging can be done using loosely-coupled to real-time distributed simulation techniques. This kind of tool naturally fosters incremental development cycles where applications or products can be integrated and tested in different versions ranging from emulated ones, to prototypes, and finally to real software and hardware, while waiting for the availability of underlying technologies. It also gives tools and methods to measure performances and to insure safety and usability.
- develop a simple model/classification to place every work and concept where it belongs and good practices to integrate them within the emulation toolkit. It may improve interactions between the different communities and foster collaboration. In the same areas, related-metrics could be better positioned and their impact evaluated.

Our experience on such topics shows that it is not at first an easy path, but that is it promising. What works for individual areas of research and for architecture or engineering should prove highly potent when dealing with the combination of them. Designing and developing an efficient distributed emulation layer and providing methods to optimize communications and relations between elements of the simulations are the core technical issues but we can probably imagine the public impact of a scenario where somehow the Sims are playing in a Doom3 world (with its PDAs) over an ns2 emulated network against human users?
We are therefore proposing the deployment of a large scale community test-bed where everyone could participate and interact with others' researches in a real-time fashion, a PCBone (Pervasive Computing Bone). This could be piloted by a taskforce, that would be in charge of defining the tools (Open Source whenever possible) and standards for emulating the future with/for Pervasive Computing technologies and contributing to monitor and inform researchers and potential users on progress made by the community.

---

[1] Vertically meaning across abstraction layers and Computer Science fields, and horizontally across large number of possible simulated objects

6. References

[1] Smart Space: http://www.nist.gov/smartspace/
[2] Easy Livinghttp://research.microsoft.com/easyliving/
[3] Changing places: http://architecture.mit.edu/house_n/index.html
[4] inHaus: http://www.inhaus-duisburg.de/en/
[5] Palm Os emulator: http://www.palmos.com/dev/tools/emulator/
[6] VMware: http://www.vmware.com/
[7] User Mode Linux: http://user-mode-linux.sourceforge.net/
[8] Bochs: http://bochs.sourceforge.net/
[9] NS2: http://www.isi.edu/nsnam/ns/
[10] NCTuns: http://nsl10.csie.nctu.edu.tw/
[11] NistNet: http://www-x.antd.nist.gov/nistnet/
[12] WiSe: http://www.bell-labs.com/org/wireless/wisext.html
[13] Diverse, An Open Source Virtual Reality Toolkit: http://diverse.sourceforge.net/
[14] Tangible User Interface Toolkit: http://dmg.caup.washington.edu/xmlSiteEngine/browsers/stylin/project51.html
[15] iGen: http://www.cognitiveagent.com
[16] Smart Fridge: http://www.lexicle.com/smartfridge
[17] The Dal lamp: http://www.violet.net/dal.jsp
[18] Roomba: http://www.irobot.com/consumer/
[19] Human PACMAN: http://mixedreality.nus.edu.sg/research-HP-infor.htm
[20] HLA: https://www.dmso.mil/public/transition/hla/
[21] MSI: http://msi.sourceforge.net/
[22] *Using Intelligent Agents to assess Pervasive Computing Technologies*, L. Ciarletta, V. Iordanov and A. Dima, International Conference on Intelligent Agents, Web Technology and Internet Commerce (IAWTIC 2001), 07/2001
[23] *Aroma-Wireless JiniTM Connection Technology-Based Adapters for Smart Space Devices*, A. Dima, K. Mills, L. Ciarletta et al., Technical Session 622 of Java One, San Francisco, 06/2000
[24] A conceptual Model for Pervasive Computing, L. Ciarletta and A.Dima, International Conference on Parallel Processing (ICPP 2000), Workshop on Pervasive Computing, 08/2000