# Model-Driven Methodologies for Pervasive Information Systems Development

José Eduardo Fernandes
Dep. Informática e Comunicações
Instituto Politécnico de Bragança
Bragança, Portugal
jef@ipb.pt

Ricardo J. Machado
Dep. Sistemas de Informação
Universidade do Minho
Guimarães, Portugal
rmac@dsi.uminho.pt

João Álvaro Carvalho
Dep. Sistemas de Informação
Universidade do Minho
Guimarães , Portugal
jac@dsi.uminho.pt

**Abstract**

This paper intends to introduce the concept of pervasive information systems (PIS) and the issues that arise from the software development for pervasive information systems. The model driven approach is generally described and its benefits to the software design are identified. Finally, some future directions for the usage of model driven methodologies within the development of PIS are highlighted, presenting some specific problems that nowadays that kind of methodologies have not yet been able to overcome.

**Keywords**: MDD, pervasive, information systems, software development

## 1 Introduction

Over the last years we have been assisting to a continuous growing trend for research and investment efforts in pervasive computing [19]. Benefiting from the growing of miniaturization and power of computing devices merged with improved wireless communication capabilities, information technology landscape is becoming more integrated with our real life landscape, allowing us to, probably in the present, and certainly in a near future, achieve some of the ambitions of the past regarding the integration of computing with reality, and the benefits derived from it.

Albeit the still common traditional use of computers as "distinct machines", they are being increasingly integrated (embedded) in other (non computing) devices, and consequently, not being seen or used as computing devices [6] [26].

Instead, the first look and thought goes to the larger (non computing) device as being a "smart" device. Nevertheless, the embedded computing device is present, monitoring and processing information, and giving the output that it was conceived for.

The dissemination (embedding) of computing devices, with its particular computing and communications capabilities, into everyday physical objects (non computing devices), allows a real time access to information by those embedded computing devices. New advances in information technology (smaller computing devices, wireless networked communication systems, interface devices and others) stimulated imagination on ways to deploy this technology in service to people and organizations. Location and status of people and goods can be tracked in real time, and the appearance of "smart" goods, which beyond monitoring themselves, can also independently communicate with other computing devices or systems, allows for monitoring and interaction that were not possible in the past, therefore allowing the conception of new systems [19].

Pervasive computing, initiated a little more than a decade ago based on the seminal work of Mark Weiser [40], envisions an environment in which computers will be embedded in our natural lives, assisting us in an ubiquitous manner on our everyday tasks. Pursuing this vision, pervasive computing system engineers and researchers, started to design devices and systems that technologically proved the viability of this vision.

## 2 Pervasive information systems

Proved and assessed the value of pervasive computing, pervasive (information) systems started being devised for use in organizations. Beside social concerns [34] regarding the particular characteristics of pervasive computing, and economic implications of its deployment [19], new systems are being though, aimed at improving, not just people's quality of life, but also the way business is done or even enabling new and innovative ways of carrying out business.

Museums [9], agriculture [8], restaurants [31], and health care [36] are examples of business domains that have been addressed by applications based on this kind of information technology. For example, in elder care [32] systems have been deployed, where pervasive computing improved the quality and the efficiency of care delivery to the elderly, assisting the staff in identifying people needing immediate attention, and to monitor trends, resulting in enhanced environment. Among the most relevant projects worldwide, the following can be emphasized: Active Badge [37], Active Campus [11], Aware Home [15], eClass/Classroom2000 [1], Labscape [4], iRoom [13], MediaCup [7] , PARCTab [38], PlantCare [18], Smart-Its [12]. All these projects aimed to prove the viability and usefulness of the pervasive computing vision.

The benefits and applications of pervasive computing are far from having reached an end. Other business domains, such as insurance companies or government agencies can also take benefits of pervasive computing. What was initially confined in developing technology to make pervasive computing out of a vision [20], surpassed the initially restricted frontiers to reach the development of applications for organizational domains, allowing the improvement of current business process or even to assist the development of new business models [19].

Along with all this, inevitably, we become aware of the presence flow and processing of information, not only by the individual computing devices, but also with a more deep significance, by the overall system that emerges from the interactions of all the computing devices, linking them together in a coherent fashion. We can then recognize the presence of a pervasive information system. Indeed, all these systems constitute, some form of information system. They gather, collect, process, store and produce information aimed at contributing to an organization or personal needs in order to achieve a set of well established objectives.

These pervasive information systems, composed of heterogeneous, mobile, or physically integrated (embedded) devices, capable of collecting, processing, and instantaneously communicating and interacting among them or others systems, opens the possibility of processing large quantity of information which, composing an information system, must be, in its essence, well designed, developed and deployed in order to, satisfying the requirements and exploring all the potential offered by the pervasive computing, maximize the revenue of these kind of systems. This is the way to efficiently satisfy the organizational or personal information needs in the pursuit of defined goals.

Research efforts so far have been mostly oriented, towards physical and virtual integration, interaction models, deployment, communication technologies and connectivity, and software architectures. It is important that these new pervasive systems also become subject of study from an information systems perspective.

Several pervasive computing characteristics, issues and challenges have been identified [2] [27] [20] [26]. Physical integration and spontaneous interoperation [16], quantity and heterogeneity of computing devices, services and applications the may be part at any moment of the system [10] are characteristics of pervasive systems that must be taken into account when designing PIS. They are important specially when aiming at easily reconfiguration of the application to cope with presence or absence of computing devices with distinct computing power or interface capabilities, availability of services, and to rapidly respond to business changing needs.

# 3 Model-driven software development

The high pace of technology innovation and changes, the growing complexity and the size of systems, the continued need of new or renewed systems with a short cycle time of development, puts pressure on the software development community which, on the need to cope with this reality and to bring higher quality and productivity, permanently searches for better development approaches and tools.

Model driven development (MDD) [28] [21] , a recent advance in software development, represents a new direction for software system development. It embodies a new manner of facing the development of software, essentially characterized by turning the models into the essence and focus of the development of a software system, independent of the possible platforms where they will be deployed. This change of focus, from code oriented to model oriented development, represents a significant evolution in software development, benefiting from an increase of abstraction, similar to what occurred, in the past, when through the introduction of compilers, there was a move in programming from assembly to 3rd generation languages [28].

The MDD approach allows for an abstraction of details of implementation platform technology and the use of concepts closer to the problem domain. Consequently, the models used to develop the system are easier to specify, understand and maintain, and are robust to changes of the technology adopted for system implementation [28].

Supporting this approach, widely accepted standards have emerged (and others will emerge), such as the Unified Modeling Language (UML) [25] , and the Object Management Group (OMG)'s [23] Model Driven Architecture (MDA) [24] initiative that reinforces and promotes this new direction on software development.

Also fundamental to MDD, evolution on automation technologies enables to expect that the final system may be constructed through automatic code generation and other automatic facilities. This allows a reduction or elimination of the semantic gap existing in the traditional software development, consisting by the difficulty of mapping the concepts used to express models of the applications on the implementation technologies of the platforms [28].

MDD comes to improve and accelerate the software development, achieving a higher quality of the developed systems. It is expected that MDD may bring an increase on the portability, interoperability and reuse, together with and an easier way to maintain and evolve a system.

The development of MDA tools and other related developments, along with the increasing deployment of MDD based projects, foster the movement toward model centric development [35]. Several research is being done in MDD field, from model transformation [30] [39], through separation of concerns [17] to executable models [22]. OMG contributes actively to the development of standards that helps MDD become a reality. In particular, UML (a key standard to MDA initiative) has been subject to revisions and evolutions, and its new version, 2.0 (to be ready during 2004), will represent a further impetus to MDD [29].

Model driven development is emerging as a major improvement in software development. Attending to the pervasive information systems' characteristics, and face to the new advances in software development, it is legitimate to expect that the benefits inherent to model driven development can be brought to the development of pervasive information systems. Nowadays it is possible to identify several variants of the original MDA recommendations, where we can find some specific methodological characteristics in what concerns either the process model and the modelling approach [21] [5] [3] [14] [33] [22]. However, none of these methodological approaches has explicitly dealt with the support of software design for pervasive information systems. This means that some work has to be done to adapt those recommendations to this application field.

## 4 Conclusions and future directions

Model-driven development of pervasive information systems must take into consideration aspects not only related with the global information system obtained from the cooperation of diverse, dispersed, integrated or mobile computing devices that in conjunction contribute to the achievement of the information system objectives, but also with the individual computing devices.

Model-driven development methodologies must be capable of producing pervasive information systems that are resilient to technological changes, rapidly absorbing new technology through the promotion of stability of models.

Models for pervasive information systems must be tailored to allow the dynamically construction and allocation of customized applications to heterogeneous computing devices, with different computational or interface capabilities.

Pervasive information systems can benefit from model-driven development approach to software developing, leveraging the capabilities of system composed by those computing devices. Several areas can be researched for this improvement, as for example:

- UML (or other languages) capabilities in improved model accuracy specification and its contribution towards model execution or model centric development
- Patterns and separation of concerns to enhance reuse of models and codifying best practices
- Model transformations and model-driven development methodologies concerning pervasive systems, and MDD supporting tools.

These concerns must be further researched in order to assure that model driven development actually brings all its potential benefits to pervasive information systems and their application domains.

These research aims at highlighting the attention that must be paid to information systems, instead of single computing devices, in what concerns the modeling and design of global solutions that apply pervasive computing principles. Pervasiveness is, indeed, an emerging property of information systems, and not of the individual computing devices; these are not pervasive, they are, typically, embedded in the environment or in other physical artifacts. The ubiquitous concept is associated with the services that pervasive information systems make available.

## References


[1] G. Abowd. Classroom 2000 – An experiment with the instrumentation of a living educational environment. *IBM Systems Journal, Special issue on Pervasive Computing*, vol. 38, no. 4, pp. 508–530, October 1999. Available at http://www.research.ibm.com/journal/sj/384/abowd.pdf (2 April 2004)

[2] G. Abowd, E. Mynatt. Charting Past, Present and Future Research in Ubiquitous Computing. *ACM Transactions on Computer-Human Interaction*, vol. 1, n$^{o}$. 7, March 2000, pp. 29–58.

[3] S. Ambler. Agile Model Driven Development Is Good Enough. *IEEE Software*, vol. 20, no. 25, September-October 2003, pp. 71–73.

[4] L. Arnstein, G. Borriello, S. Consolvo, C. Hung, J. Su. Labscape: A Smart Environment for the Cell Biology Laboratory. *IEEE Pervasive Computing*, July-September 2002, pp. 13–21.

[5] C. Atkinson, T. Kühne. Model-Driven Development: A Metamodeling Foundation. *IEEE Software*, vol.20, no.5, September-October 2003, pp. 36–41.

[6] G. Banavar, J. Beck, E. Gluzberg, J. Munson, J. Sussman, D. Zukowski. Challenges: An Application Model for Pervasive Computing. In *Proceedings of the 6th ACM/IEEE International Conference on Mobile Computing and Networking (Mobicom 2000)*, ACM Press, 2000, pp. 266–274.

[7] M. Beigl, H. Gellersen, A. Schmidt. MediaCups: Experience with Design and Use of Computer-Augmented Everyday Objects. *Computer Networks, Special Issue on Pervasive Computing*, Elsevier, vol. 35, no. 4, March 2001, pp. 401-409.

[8] J. Burrell, T. Brooke, R. Beckwith. Vineyard Computing: Sensor Networks in Agricultural Production. *IEEE Pervasive Computing*, January-March 2004, pp. 38–45.

[9] M. Fleck, M. Frid, T. Kindberg, E. O'Brien-Strain, R. Rajani, M. Spasojevic. From information to remembering - ubiquitous systems in interactive museums. *IEEE Pervasive Computing*, April-June 2002, pp. 13–21.

[10] R. Grimm, J. Davis, B. Hendrickson,E. Lemar, A. MacBeth,S. Swanson, T. Anderson, B. Bershad, G. Borriello, S. Gribble, D. Wetherall. Systems directions for pervasive computing. In *Proceedings of the Eighth Workshop in Operating Systems (HotOS-VIII)*, Elmau, Germany, May 2001, pp. 147–151.

[11] W. Griswold, P. Shanahan, S. Brown, R. Boyer, M. Ratto, R. Shapiro, T. Truong. ActiveCampus - Experiments in Community-Oriented Ubiquitous Computing. *UCSD CSE Technical Report CS2003-0750*, 2003.

[12] L. Holmquist, H-W. Gellersen, G. Kortuem, A. Schmidt, M. Strohbach, S. Antifakos, F. Michahelles, B. Schiele, M. Beigl, R. Mazé. Building Intelligent Environments with Smart-Its. *IEEE Computer Graphics*, January-February 2004, pp. 56–63.

[13] B.Johanson, A. Fox, T. Winograd. The Interactive Workspaces Project: Experiences with Ubiquitous Computing Rooms. *IEEE Pervasive Computing*, April-June 2002, pp. 71–78. Also available at http://swig.stanford.edu/pub/publications/iwork-overview-layout.pdf (2 April 2004).

[14] J. Jorgensen, C. Bosse. Executable Use Cases: Requirements for a Pervasive Health Care system. IEEE Software, March-April 2004, pp. 34–41.

[15] C. Kidd, R. Orr, G. Abowd, C. Atkeson, I. Essa, B. MacIntyre, E. Mynatt, T. Starner, W. Newstetter. The Aware Home: A Living Laboratory for Ubiquitous Computing Research. In *Proceedings of Second International Workshop on Cooperative Buildings 1999*, Editors, Streitz, J. Siegel, V. Hartkopf, S. Konomi, Pittsburgh. LNCS 1670. Springer: Heidelberg, Pittsburgh, PA, October 1999. Also available at http://www.cc.gatech.edu/fce/ahri/publications/cobuild99_final.PDF (01 April 2004)

[16] T. Kindberg, A. Fox. System Software for Ubiquitous Computing. IEEE Pervasive Computing, January-March 2002, pp. 70–81.

[17] V. Kulkarni, S. Reddy. Separation of Concerns in Model-Driven Development. IEEE Software, vol. 20, no. 5, September-October 2003

[18] A. LaMarca, W. Brunette, D. Koizumi, M. Lease, S. Sigurdsson, K. Sikorski, D. Fox, G. Borriello. PlantCare: An Investigation in Practical Ubiquitous Systems. G. Borriello and L.E. Holmquist (Eds.), *UbiComp 2002, LNCS 2498*, pp. 316–332, 2002. Springer-Verlag Berlin Heidelberg 2002

[19] M. Langheinrich, V. Coroama, J. Bohn, M. Rohs. As we may live - Real world implications of ubiquitous computing. Technical Report, Institute of Information Systems, Swiss Federal Institute of Technology, Zurich, Switzerland, 2002. http://www.inf.ethz.ch/vs/publ/papers/uc-implications.pdf (13 July 2003)

[20] K. Lyytinen, Y. Yoo. Issues and Challenges in Ubiquitous Computing. Communications of the ACM, vol.45, n$^{o}$12, December 2002, pp. 63–65.

[21] S. Mellor, A. Clark, T. Futagami. Model-Driven Development. *IEEE Software*, vol. 20, no. 5, Setember-October 2003, pp. 14–18.

[22] S. Mellor, M. Balcer. Executable UML - A Foundation for Model Driven Architecture. Addison-Wesley, 2002.

[23] Object Management Group. http://www.omg.org (13 de Julho de 2003)

[24] Object Managment Group. MDA Guide Version 1.0.1. June 2003. http://www.omg.org/docs/omg/03-06-01.pdf (13 July 2003)

[25] Object Managment Group. Unified Modeling Language v1.5. March 2003. http://www.omg.org/docs/formal/03-03-01.pdf (13 July 2003)

[26] D. Saha, A. Mukherjee. Pervasive Computing: A Paradigm for the 21st Century. *IEEE Computer, vol. 36*, n$^{o}$ 3, March 2003 , pp. 25–31.

[27] M. Satyanarayanan. Pervasive Computing: Vision and Challenges. *IEEE Personal Communications*, August 2001, pp. 10–17.

[28] B. Selic. The Pragmatics of Model-Driven Development. *IEEE Software*, vol. 20, no. 5, September-October, 2003 , pp. 19–25.

[29] B. Selic. UML 2.0: Exploiting Abstraction and Automation. *SD-Times (Editorials and Opinions)*, Issue 98, 15 March 2004 Available at: http://www.sdtimes.com/opinions/guestview_098.htm. (14 April 2004)

[30] S. Sendall, W. Kozaczynski. Model Transformation: The Heart and Soul of Model-Driven Software Development. *IEEE Software*, Vol. 20 , No. 5 , September-October 2003, pp. 42–45.

[31] V. Stanford. Pervasive Computing Puts Food on the Table. IEEE Pervasive Computing, January-March 2003, pp. 9–14.

[32] V. Stanford. Using Pervasive Computing to Deliver Elder Care. IEEE Pervasive Computing, January-March 2003, pp. 10–13.

[33] L. Star. Executable UML: How to Build Class Models. Prentice Hall, 2002.

[34] A. Stone. The Dark Side of Pervasive Computing. *IEEE Pervasive Computing*, January-March 2003, pp. 4–8.

[35] A. Uhl. Model Driven Architecture Is Ready for Prime Time. *IEEE Software*, vol. 20, no. 5, September-October 2003, pp. 70–72.

[36] U. Varshney. Pervasive Healthcare. *IEEE Computer*, December 2003, pp. 138–140.

[37] R. Want, A. Hoper. Active Badges and Personal Interactive Computing Objects. *IEEE Transactions on Consumer Electronics*, vol. 38, Nn. 1, February 1992.

[38] R. Want, B. Schilit, A. Norman, R. Gold, K. Petersen, D. Goldberg, J. Ellis, M. Weiser. An overview of the PARCTAB ubiquitous computing experiment. *IEEE Personal Communications*, vol. 2 , issue 6 , December 1995 pp. 28–43.

[39] T. Weis, A. Ulbrich, K. Geihs. Model Metamorphosis. *IEEE Software*, vol. 20, no.5, September-October 2003, pp. 46–81.

[40] M. Weiser. The Computer for 21st Century. *Scientifc American*, vol. 265, no. 3, September 1991, pp. 94–104.