

Research Article

Rochdev: A Humanistic Development Approach

Taoufik Ben Hassine* 

School of Computer Science, Manouba University, Manouba, Tunisia

Abstract

Given the difference in culture of developers in different regions of the world and in order to allow a coexistence of these cultures at the same time as a contribution of all these developers in development projects which may consist of collaboration in the same project or cooperation in separate projects, this article presents the Rochdev development approach, which proposes a theoretical, horizontal, common development culture shared by all vertical, specific and alternative development cultures. Rochdev stands out for its humanism which consists of respecting the different cultures of IoT (Internet of Things) development without preventing co-development based on the sharing of a super-culture made up of a collection of abstract concepts which is not there to replace the different cultures but which allows us to understand the culture of the other which represents an alternative to its culture good to know and why not learn to enrich its own. This collection of high-level concepts is grouped in a meta-model called "MODIDO". Rochdev is based on an IoT language called "LIDO". "LIDO" have "MODIDO" as an abstract syntax and a textual notation called "NTIDO". This article describes the theoretical aspect of the approach. The application of Rochdev to the KIWI cultivation area in Italy is an example of the practice of this approach in a specific agricultural field.

Keywords

Rochdev, Humanistic Development Approach, Horizontal Culture, Vertical Culture, Collection of Concepts, LIDO, MODIDO

1. Introduction

Successful development, regardless of the targeted area, is a very important achievement for the emancipation of women, men, and peoples. Moreover, some, rightly or wrongly, classify countries into developed, developing, and underdeveloped countries. In some cases, development is not accompanied by an approach or methodology but is carried out in a hybrid manner without tying in or conforming to a framework that at least refers to confirmed knowledge, know-how, and management skills. As a result, the development process turns out to be anarchic or even chaotic. To succeed in this development, it must be carried out rationally. Rational development is development based on knowledge of what we want to develop, knowledge of the future state to which this devel-

opment will lead, and the human, financial, and regulatory resources to implement to achieve it. An approach is recommended, even necessary, to accompany and guide this development. This approach can be slightly or deeply close or, on the contrary, distant and removed from the details of development. An approach is caught between a global and general vision based on vague ideas and a precise approach that follows the course of development step by step. An approach can ensure a vision, or a vision accompanied by an approach. The choices of the scope and consistency of the approach are left to the responsible men charged with leading this development. This article describes a development approach called Rochdev [1, 2] which is based on the

*Corresponding author: taoufik.benhassine@ensi.mu.tn (Taoufik Ben Hassine)

Received: 30 July 2025; Accepted: 22 August 2025; Published: 23 September 2025



Copyright: © The Author(s), 2025. Published by Science Publishing Group. This is an **Open Access** article, distributed under the terms of the Creative Commons Attribution 4.0 License (<http://creativecommons.org/licenses/by/4.0/>), which permits unrestricted use, distribution and reproduction in any medium, provided the original work is properly cited.

knowledge of the reality that one wants to develop, on the knowledge of the reality transformed following the development and, on the transformations, to be made. Rochdev is a contraction of “Development Approach.” Rochdev adopts, blends, and combines domain-driven design, model-driven engineering in general, and model-driven architecture in particular. It includes a theoretical and conceptual aspect, a sort of vision of how to conduct development regardless of the targeted domain. Rochdev also includes a practical aspect applied to the IoT field. The rest of this article is divided into three sections. Section 2 discusses the theoretical aspect of Rochdev, which is intended to be horizontal and universal. Section 3 presents the practical aspect of Rochdev as applied to KIWI cultivation in Italy. Section 4 concludes this article.

2. The Theoretical Aspect of Rochdev

A development concerns one (or more) domain(s). According to Rochdev, who refers to domain-directed design [12], it follows that an in-depth study of this domain is necessary. Rochdev advises building up a multimedia documentary fund that helps to carry out this study. Rochdev, referring to model-directed engineering [13, 15], stipulates that the product of this study must be a meta-model of the domain studied. A meta-model retains a skeleton and a foundation of the domain expressed in higher-level concepts. It models the constants of the domain and ensures the sustainability of

knowledge of it, avoiding redesign efforts when things change. This meta-model must comply with a meta-meta-model which is for the OMG (Object Management Group) the MOF (Meta Object Facility) [6]. Rochdev recommends basing itself on standards generally coming from the OMG. This domain meta-model is used to generate a model of the application system by derivation. We can say that we transform the domain meta-model by subtraction or derivation into a simple restricted model targeted by the application system to be implemented. The meta-model and the model are both digitally neutral. That said, they do not include any computer or digital concepts. The application system model represents, according to the OMG Model-Directed Architecture [11, 14], a digitization-independent model or CIM (Computational Independent Model), which represents an analysis model. This CIM will be transformed into a PIM (Platform Independent Model), which is nothing other than a non-targeted design model, i.e. one that has no preference for a given target platform. From a PIM, one can generate, by transformation, and depending on the targeted platforms, one or more PSMs (Platform Specific Models). Each platform has its own digital culture, and the PIM is there to erase the boundaries of incompatibility or misunderstanding since it is designed on a base of commonly shared concepts. The PSM will result in code in Java, C++, Python, or other languages. The following diagram (Figure 1) shows the approach taken by Rochdev.

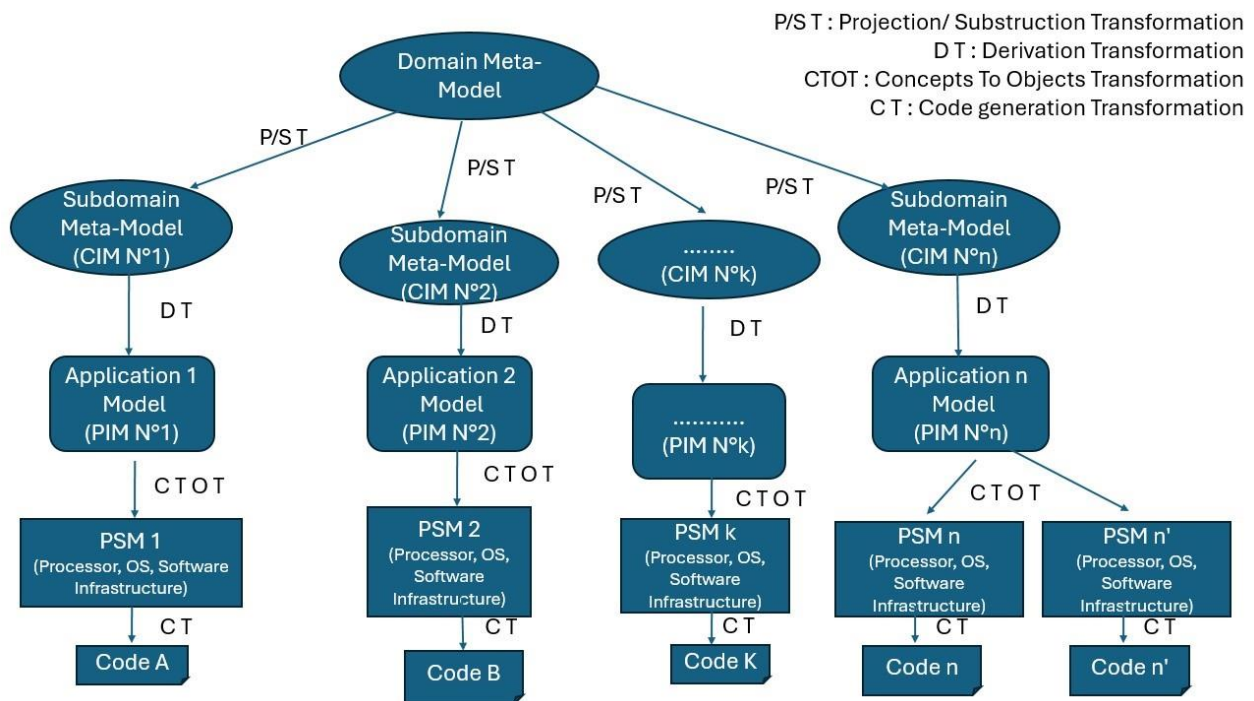


Figure 1. Rochdev Transformation Models.

3. A Practical Case from Rochdev: KIWI Cultivation

As shown in the previous section, Rochdev can be applied to all fields. This practical case discussed in this section concerns the agricultural sector and more specifically kiwi cultivation in Italy. Italy is the second largest kiwi producer after China with a saleable production of approximately 448,000 tons in 2014. The two countries, along with New Zealand, Chile, and Greece, account for 93% of global kiwi production. FAMOSA is an Italian company that works on technological support for agriculture, offering solutions for crop monitoring and management. They have developed

wireless sensor networks, based on “Libelium” technology, in a kiwi plantation with GPRS and Sigfox to develop precise irrigation strategies for farmers. It can contain 6 different radio interfaces: 2.4 GHz WiFi, 5 GHz WiFi, 3G/GPRS, Bluetooth, XBee, and LoRa. In addition, “Meshlium Gateway” allows for the integration of various communication and information and data processing materials. It can therefore integrate a GPS module for mobile applications and vehicles and be powered by solar energy and batteries. These features, along with an IP65 aluminum housing, allow “Meshlium Gateway” to be placed anywhere outdoors. Figure 2 shows the different components of this integrated solution.

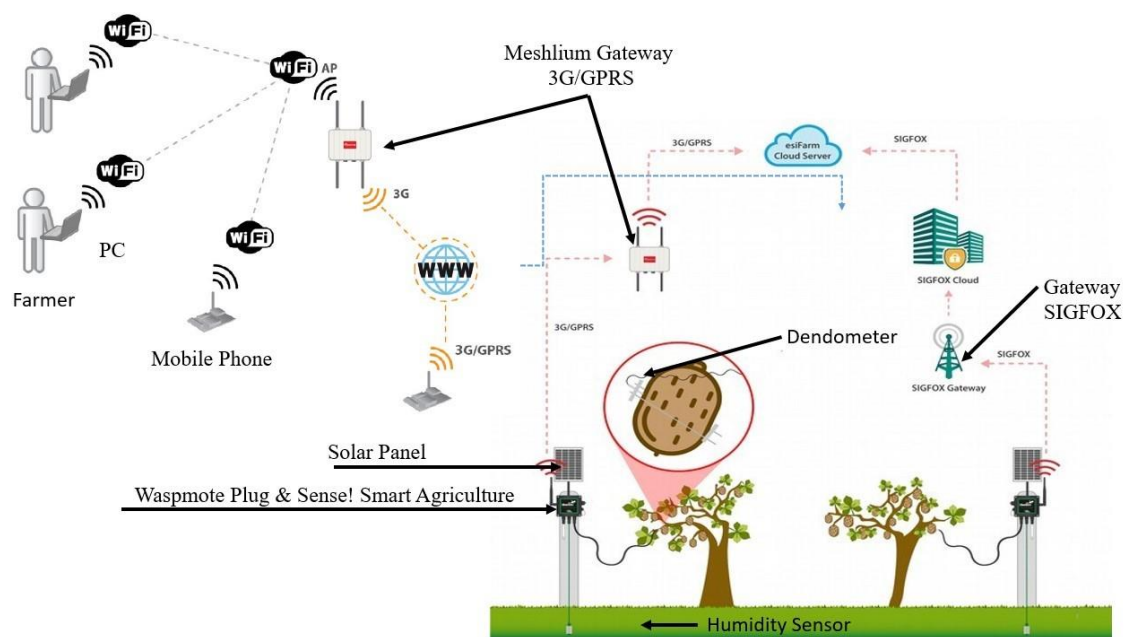


Figure 2. Meshlium Gateway Integration for KIWI Solution [4].

The technical solution based on “Libelium” hardware is based on the 3G/GPRS communication bridge and the “Cloud Server” equipped with this bridge and interfacing with SigFox. The “Cloud Server” thus constitutes an integrator of different communication networks.

The goal of the KIWI project is analogous in its principles to the MediaCup project. The goal of developing the MediaCup project [5] was to augment an ordinary coffee cup with sensing, processing power, and communication capabilities (integrated in the cup’s bottom). It’s a ubiquitous computing project [7]. Intimacy is something that relates to our innermost selves, something personal, closely felt. Such a construction could include love, closeness, or spirituality. Or perhaps it is in the way we understand, feel and talk about our lives, our bodies, our identities, our souls. In all these ways, intimacy transcends technology and has a role to play in shaping. Having said that, there has been an idea of intimate computing for as long as there has been a vision of ubiquitous computing

[9, 10]. The two are inexorably linked in the pages of the September 1991 issue of Scientific American [8]. In that month’s issue of the magazine, Mark Weiser, articulated his vision of ubiquitous computing – “we are trying to conceive a new way of thinking about computers in the world, one that takes into account the natural human environment and allows computers themselves to vanish into the background” [8]. In fact, the computer that we know is a primary artefact. As secondary artefacts, computers will ideally be invisible and interacted with implicitly. So, in a certain sense, computers will disappear, while *digital artefacts* will emerge, encompassing all sorts of devices and things that are not computers themselves but augmented with the ability to process and exchange digital information [5]. In their seminal paper Intimate (Ubiquitous) Computing, Genevieve Bell, Tim Brooke, Elizabeth Churchill, and Eric Paulos write that: “As appliances and computation move away from the desktop, and as designers move toward designing for emotion and social

connection rather than usability and utility, we are poised to design technologies that are explicitly intimate and/or intimacy promoting.” [9].

The KIWI project aims to let the KIWI culture intimacy and ubiquitous to the fruit and the farmer. That made Rochdev a humanistic development approach. Figure 3 shows the MediaCup.



Figure 3. Mediacup is an ordinary coffee cup augmented with information capture, computational, and communication capabilities. [5]

Otherwise, our IoT-specific modeling language, LIDO [3], is at the heart of the ROCHDEV development approach we are experimenting with. The two facets of the LIDO language, namely the MODIDO and NTIDO meta-models, representing respectively its abstract syntax and concrete syntax (Figure 4),

will support the modeling of the IoT solution that a developer wishes to implement.

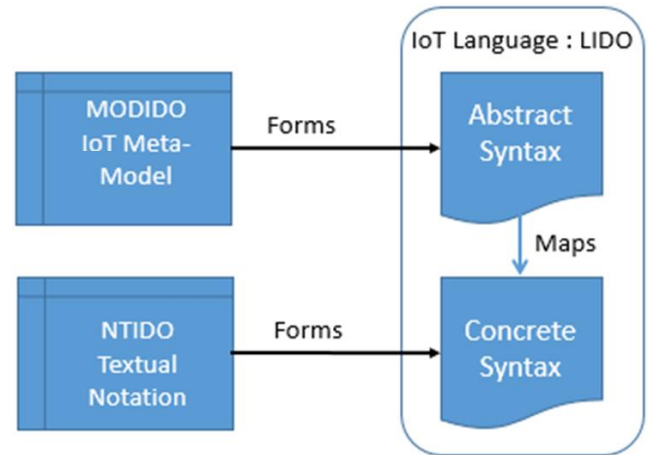


Figure 4. MODIDO and NTIDO component of LIDO.

The approach is accompanied by a development process that requires a developer to first derive MODIDO to create a model of their solution and then write this solution using the NTIDO textual notation. The technical support of the approach that is the INDIT integrated development environment (IDE) will allow the developer to edit his solution and generate skeleton code of what he will undertake to implement his solution. Figure 5 describes this demarch.

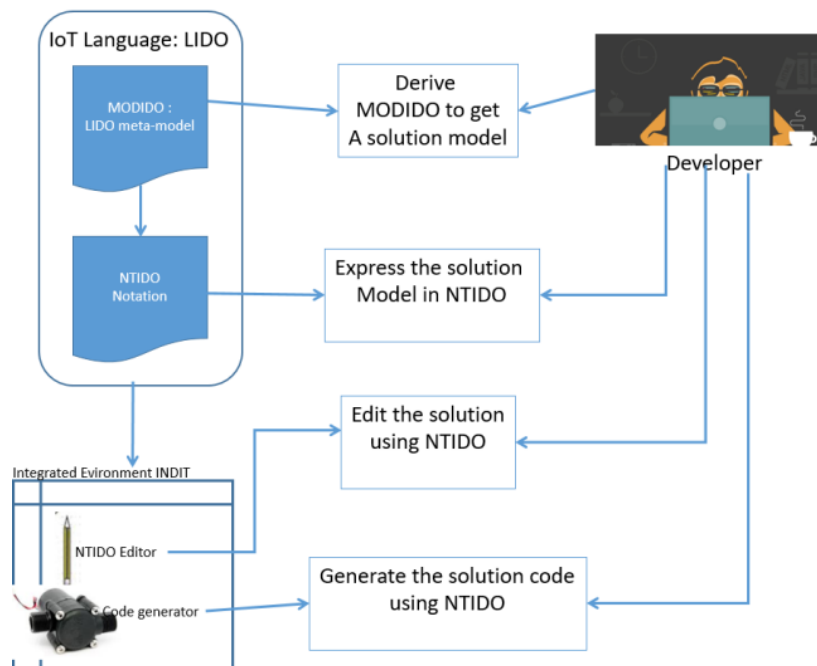


Figure 5. The demarch: From the LIDO language to the INDIT environment.

4. Conclusion

Our “Rochdev” approach based on domain model-driven engineering and model and transformation architecture has enabled us to support an IoT application for an agricultural sector dealing with KIWI cultivation in Italy, the second largest producer of this fruit in the world after China. The strength of the LIDO language and its MODIDO meta-model has made it possible to strengthen and improve the practical aspect of Rochdev, which is now considered a universal approach. Rochdev is universal because it has a theoretical aspect that shares a collection of concepts detached from the realities that its practical aspect addresses and supports. The Italian KIWI, competing with the Chinese KIWI, can be considered a global and universal KIWI if we only consider the theoretical aspect of Rochdev and can be considered a local and specific KIWI if we look at the practical side of the approach. LIDO and MODIDO constitute the pivot and the spirit of the Rochdev approach. Their universality induces that of Rochdev. The universality of Rochdev lies in its horizontality. Indeed, Rochdev offers the possibility of approaching a field with just concepts that do not depend in any way on the culture of its engineers or developers. Rochdev offers a super-culture or a meta-culture that makes IoT a raw material, a kind of rock that can be sculpted to arrive at a specific work, a specific and vertical culture that enriches and participates with its peers in the constitution of horizontal cultures for all the humans and the entire humanity.

Abbreviations

CIM	Computational Independent Model
IoT	Internet of Things
MOF	Meta Object Facility
OMG	Object Management Group
PIM	Platform Independent Model
PSMs	Platform Specific Models

Author Contributions

Taoufik Ben Hassine is the sole author. The author read and approved the final manuscript.

Conflicts of Interest

The authors declare no conflicts of interest.

References

- [1] Ben HASSINE Taoufik, KHAYATI Oualid and Ben GHEZALA Henda. “An IoT domain meta-model and an approach to software development of IoT solutions”. In: 2017 International Conference on Internet of Things, Embedded Systems and Communications (IINTEC). IEEE, 2017. p. 32-37, <https://doi.org/10.1109/IINTEC.2017.8325909>
- [2] Taoufik Ben Hassine, “Approach to building solutions for the Internet of Things”, Our Knowledge Publishing, September 2022, ISBN: 978-620-5-16059-6.
- [3] Taoufik Ben Hassine, “LIDO: A Universal Language”, World Journal of Advanced Research and Reviews, 2023, 18(02), 361–364, <https://doi.org/10.30574/wjarr.2023.18.2.0826>
- [4] Taoufik Ben Hassine. “A Language & an Approach for the Development of IoT Solutions”. American Journal of Electrical and Computer Engineering. Vol. 6, No. 1, 2022, pp. 1-14. <https://doi.org/10.11648/j.ajece.20220601.11>
- [5] Beigl, M., Gellersen, H.-W. and Schmidt, A. MediaCup. Experience with Design and Use of Computer-Augmented Everyday Objects. Computer Networks, Vol. 35, No. 4, Special Issue on Pervasive Computing, Elsevier, March 2001, pp. 401-409. https://doi.org/10.1007/3-540-48157-5_30
- [6] Tang, W. (2009). Meta Object Facility. In: LIU, L., ÖZSU, M. T. (eds) Encyclopedia of Database Systems. Springer, Boston, MA. https://doi.org/10.1007/978-0-387-39940-9_914
- [7] Weiser Mark. Gold R. The origins of ubiquitous computing research at Park in the late 1980s, IBM Systems Journal (1999).
- [8] Weiser, M. The Computer for the Twenty-first Century, Scientific American, 265, 3, pp. 94-10, 1.
- [9] Bell, Genevieve & Brooke, Tim & Churchill, Elizabeth & Paulos, Eric. (2003). Intimate ubiquitous computing10.
- [10] Reeves, Stuart. (2012). Envisioning ubiquitous computing. Conference on Human Factors in Computing Systems - Proceedings. <https://doi.org/10.1145/2207676.2208278>
- [11] Korongo, Julia & Mbugua, Samuel & Mbuguah, Samuel. (2022). A Review Paper on Application of Model-Driven Architecture in Use-Case Driven Pervasive Software Development. International Journal of Computer Trends and Technology. 70. 19-26. <https://doi.org/10.14445/22312803/IJCTT-V70I3P104>
- [12] Ozan Özkan, Önder Babur, Mark van den Brand, Domain-Driven Design in software development: A systematic literature review on implementation, challenges, and effectiveness, Journal of Systems and Software, Volume 230, 2025.
- [13] Alberto Rodrigues da Silva, Model-driven engineering: A survey supported by the unified conceptual model, Computer Languages, Systems & Structures, Volume 43, 2015, Pages 139-155, <https://doi.org/10.1016/j.cl.2015.06.001>

- [14] Julia N. Korongo, Samuel T. Mbugua, Samuel M. Mbuguah, "A Review Paper on Application of Model-Driven Architecture in Use-Case Driven Pervasive Software Development," *International Journal of Computer Trends and Technology (IJCTT)*, vol. 70, no. 3, pp. 19-26, 2022. Crossref, <https://doi.org/10.14445/22312803/IJCTT-V70I3P104>
- [15] V. V. Graciano Neto *et al.*, "Model-Driven Engineering Ecosystems," *2019 IEEE/ACM 7th International Workshop on Software Engineering for Systems-of-Systems (SESoS) and 13th Workshop on Distributed Software Development, Software Ecosystems and Systems-of-Systems (WDES)*, Montreal, QC, Canada, 2019, pp. 58-61, <https://doi.org/10.1109/SESoS/WDES.2019.00016>