

## **ODBMS.ORG User Report No.3/08**

Editor Roberto V. Zicari- [www.odbms.org](http://www.odbms.org)

July 2008.

**Category: Academia**

**Domain:** Research/Education

**User Name: Philippe Roose**

**Title:** Associate Professor

**Organization: LIUPPA/IUT de Bayonne, France.**

Philippe Roose is Associate Professor at the IUT of Bayonne and member of the LIUPPA (Laboratoire d'Informatique de l'Université de Pau et des Pays de l'Adour) - France since 2000. He received his PhD in 2000 called "/ELKAR : A Component Based Re-engineering Methodology and Architecture to Provide Cooperation". He is now working on Software Component and Architectures for Distributed Multimedia Applications with QoS management.

*Q1. Please explain briefly what are your application domains and your role in the enterprise.*

**Philippe Roose:** I work for a public research laboratory in France called LIUPPA in computer science. I lead 2 research projects dealing with software architecture for wireless sensor networks, and more generally, for constrained peripherals.

*Q2. When the data models used to persistently store data (whether file systems or database management systems) and the data models used to write programs against the data (C++, Smalltalk, Visual Basic, Java, C#) are different, this is referred to as the "impedance mismatch" problem. Do you have an "impedance mismatch" problem?*

**Philippe Roose:** We had... and we decided to solve it using a common language as Java. This can be done because with research projects we are not linked to technologies, I presume that it cannot be done with industrial projects, or project based on re-use or re-engineering.

When we had such an "impedance mismatch" problem, we developed specific connectors called "active rules". Their jobs was to make the link (technically and semantically) between the persistent source(s) and the destination one(s).

*Q3. What solution(s) do you use for storing and managing persistence objects? What experience do you have in using the various options available for persistence for new projects? What are the lessons learned in using such solution(s)?*

**Philippe Roose:** All our objects are 'serializable'. We send them and store them into a middleware. In fact, we do not manipulate exactly objects but components. We store them into software components repository. Then each peripheral as Sensor, PDA, PC host a 'part' of a middleware. This software manages the peripheral and decides if some local component has to be stored and transferred or is a stored component as to be migrated on the peripheral itself. Is the request are sent to the component repository in order to get the serialized instance.

With some sensors (like Sun Spot), it cannot be done because of the lack of class Loader mechanisms. Do we duplicate components on each sensor. The persistence is only based on the state of the object instead of the object itself.

*Q4. Do you believe that Object Database systems are a suitable solution to the "object persistence" problem? If yes why? If not, why?*

**Philippe Roose:** Yes, It can, it will! The most interesting thing for us is the use of polymorphism. But for the moment, we cannot use OODBMS as we should. Such solution is too heavy to run on constraints peripherals. But as they power capacities increase, and as OODBMS solution are improved, we plan to use OODBMS in a short future.

*Q5. What would you wish as new research/development in the area of Object Persistence in the next 12-24 months?*

**Philippe Roose:** Light weight engine.