Table of Contents

P@J (or PatJ) ............................................................................................................................................................................ 3
P@J (or PatJ)

Although object-oriented languages are nowadays the mainstream for application development, several research contexts suggest that a multi-paradigm approach is worth pursuing.

In particular, a declarative, logic-based paradigm could fruitfully add functionalities related to intelligence, adaptivity, and conciseness in expressing algorithms. The aim of the P@J project is to provide a framework for enhancing interoperability between Java and Prolog, based on the tuProlog open-source Prolog engine for Java. P@J achieves smoother language-interoperability through two stacked layers:

- an API layer for automated mapping of Java types into Prolog types (and viceversa) and seamless exploitation of the Generic Collections Framework;
- an annotation layer, that aims at truly extending Java programming with the ability of specifying Prolog-based declarative implementations of Java methods, relying on Java annotations.

Click [here](#) for downloading the latest version of P@J!

Check the articles describing the features of P@J [here](#)