

A modular aspect-oriented programming approach of join point interfaces

Vidal C.

Madariaga E.

Jiménez C.

Carter L.

This paper describes and analyzes the main differences and advantages of the Join Point Interfaces (JPI) as an Aspect-Oriented Programming (AOP) approach for the modular software production concerning the standard aspect-oriented programming methodology for Java (AspectJ) to propose a structural modeling approach looking for modular software solutions. Using a Software Engineering point-of-view, we highlight the relevance of structural and conceptual design for JPI software applications. We model and implement a classic example of AOP using AspectJ and JPI as an application example to review their main difference and highlight the JPI consistency between products (models and code). Our proposal of UML JPI class diagrams allows the definition of oblivious classes which know about their JPI connections, an essential element to adapt and transform tradition like-AspectJ AOP solutions to their JPI version. Thus, for the modular software production and education, JPI seems an ideal software development approach. © 2018 The Science and Information (SAI) Organization Limited.

Aspect-Oriented Programming

AspectJ

Class diagrams

JPI

UML