

PROSE – A Platform for Dynamic AOP

Angela Nicoară, Gustavo Alonso
 Information and Communication Systems, ETH Zürich
 IP8: System and Software Architecture

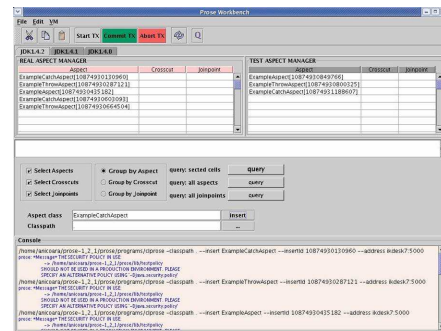
Introduction

- A prerequisite for both fixed and mobile computing is **the ability to adapt**: to different computing environments or to external changes occurred in their current computing environment. Otherwise the services are forced to be foreseen with everything they may possibly need during their operational life time.
- Recent developments in service architectures suggest that dynamic adaptations could be implemented with dynamic AOP.
- **PROSE (PROgrammable extenSions of sERVICES)** holds the promise of allowing orthogonal adaptations at run-time.
- **PROSE** supports three *prototypes*: one based on the debugger interface of the JVM (JVMDI-based weaver) and other two based on JIT compiler technology (*stub* and *advice weavers*).
- The **key problem** for dynamic AOP techniques is the potential overhead that might be inflicted on the application => **PROSE advice weaver**, a modular and flexible architecture intended to improve the efficiency of the dynamic AOP process, using *method code replacement*.

Features

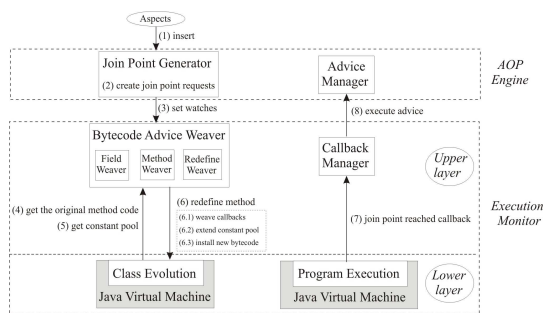
- **PROSE** allows to modify the behavior of a running application:
 - identify *where* to change the program (potential points of interest may be, e.g., method boundaries (method entries and exits), method redefinition, variable access or exception throws and catches)
 - establish *what* additional actions are needed at those points
- **PROSE** supports run-time weaving and unweaving of aspects

PROSE workbench



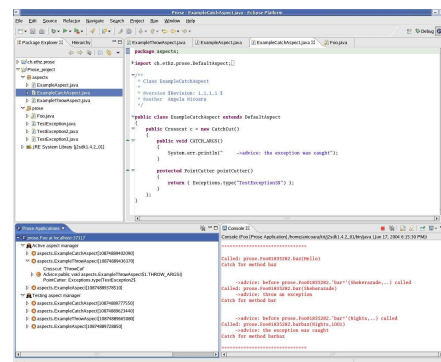
- distributed visualization of remote aspects
- run-time monitoring of aspects
- the monitor tool displays information about:
 - what aspects are currently inserted in the system
 - what join-points belong to each aspect
 - what are the join-points of the system

PROSE advice weaver architecture



- the *AOP Engine* accepts aspects (1) and transforms them into join-point requests (2)
- it activates the join-point requests by invoking methods of the *Execution Monitor* (3)
- the *Bytecode Advice Weaver* module is responsible for the bytecode manipulations and the methods instrumentation (4) - (6)
- when the program execution reaches one of the activated join-points (7), the *Execution Monitor* notifies the *AOP Engine* which then executes the advice (8)

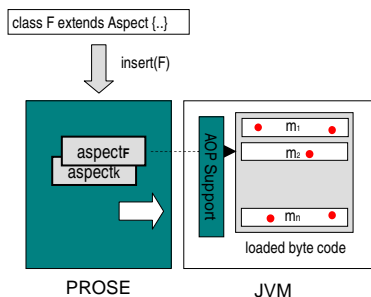
PROSE as a plugin for Eclipse



PROSE is used for:

- *spontaneous container networks*, where transactional interaction, access control, and container managed persistence are dynamic extensions that can be added to or removed from a mobile application at will
- *hot fixes in web services*
- *adapting the behavior of robots* (RCX-based)

PROSE design



- add aspect F to PROSE
- register stop requests (JVMDI)
- maintain the mapping between aspects and join-points

PROSE website: <http://prose.ethz.ch>
PROSE available for download: <http://sourceforge.net/projects/jprose>

