Handles

Property-Propagating First Class Reference for Dynamically-Typed Languages.

Jean-Baptiste ARNAUD
jean-baptiste.arnaud@inria.fr
RMoD, INRIA Lille Nord Europe
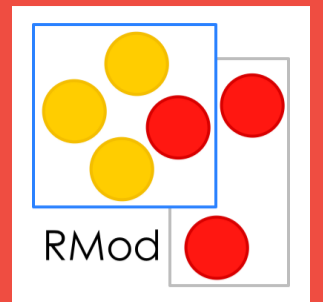
Scientific challenge: Build a dynamic (but secure) language

Controlling references and giving specific semantics to single objects and to graphs of objects is essential to be able to build more secure systems, but is notoriously hard to achieve in absence of static type systems. This is due to tree constraints:

- Open world, system can potentially change at runtime.
- No type information until runtime.
- Presence of reflection.

Problem: what is the underlying mechanism that can support the definition of properties (such as revocable, read-only, lent...) at the reference level in the absence of a static type system?

Solution

Our solution is the Handle:
Handles are first class references (i.e., objects modelling a reference).
The Handle provides:

- Indistinguishable property: Handle represent another object (target object) of the system, from the system point of view is considered as the target object.
- State guarding: it retains its own version of the state of the target object.
- Shadow class: it can change the behavior of the target object (only for this reference).
- Handle propagates its own configuration to the target object graph, when accessed.

Metahandle provides a real time control of Handles

- Change the configuration of Handle at runtime (Behavior change, State guarding, etc.).
- Install the Handle (remove the Handle; if state guarding is activate install the current state on the target object).
- Can be created only at Handle creation time.
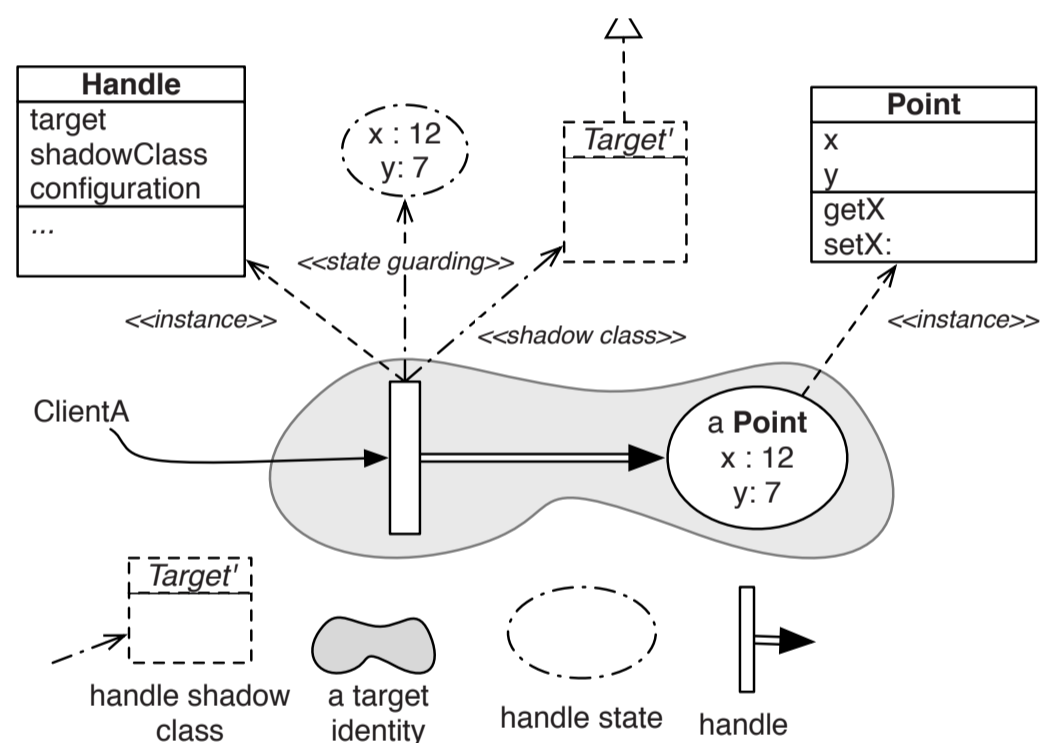
Result

A working implementation with a specific virtual machine and API for using Handles, do on Pharo language.

Implementation of complex security properties (such, Read-Only, Membrane, Mirror, Software Transactional Memory, etc.)

- With a minimalist design (less than 5 classes and 30 methods).
- Usable performance.

Do you want to know more?

http://jeanbaptiste-arnaud.eu/handles

Jean-Baptiste Arnaud, Marcus Denker, Stéphane Ducasse, Damien Pollet, Alexandre Bergel and Mathieu Suen. *Read-Only Execution for Dynamic Languages*. In Tools Europe. Springer Verlag, June 2010.