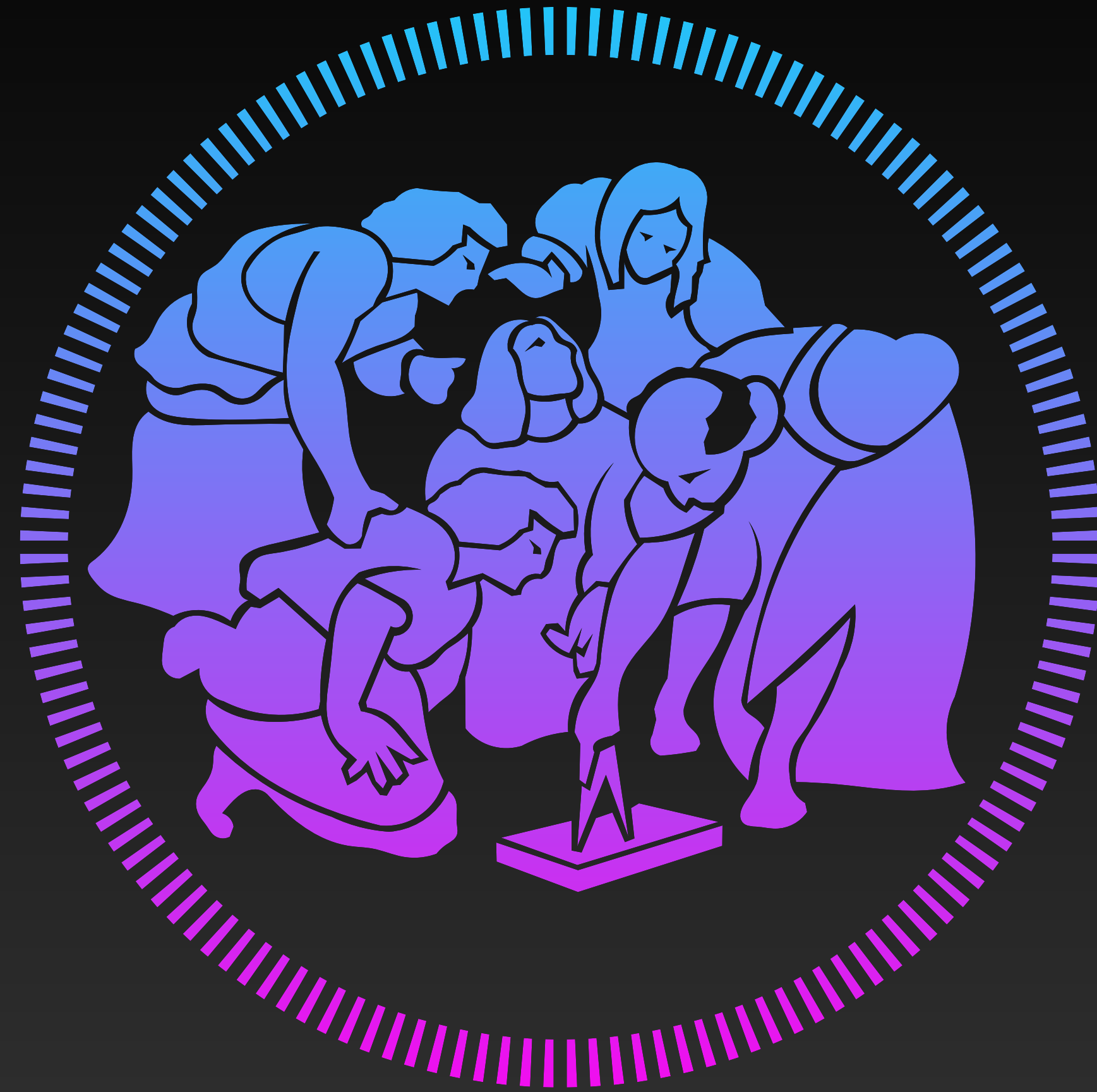
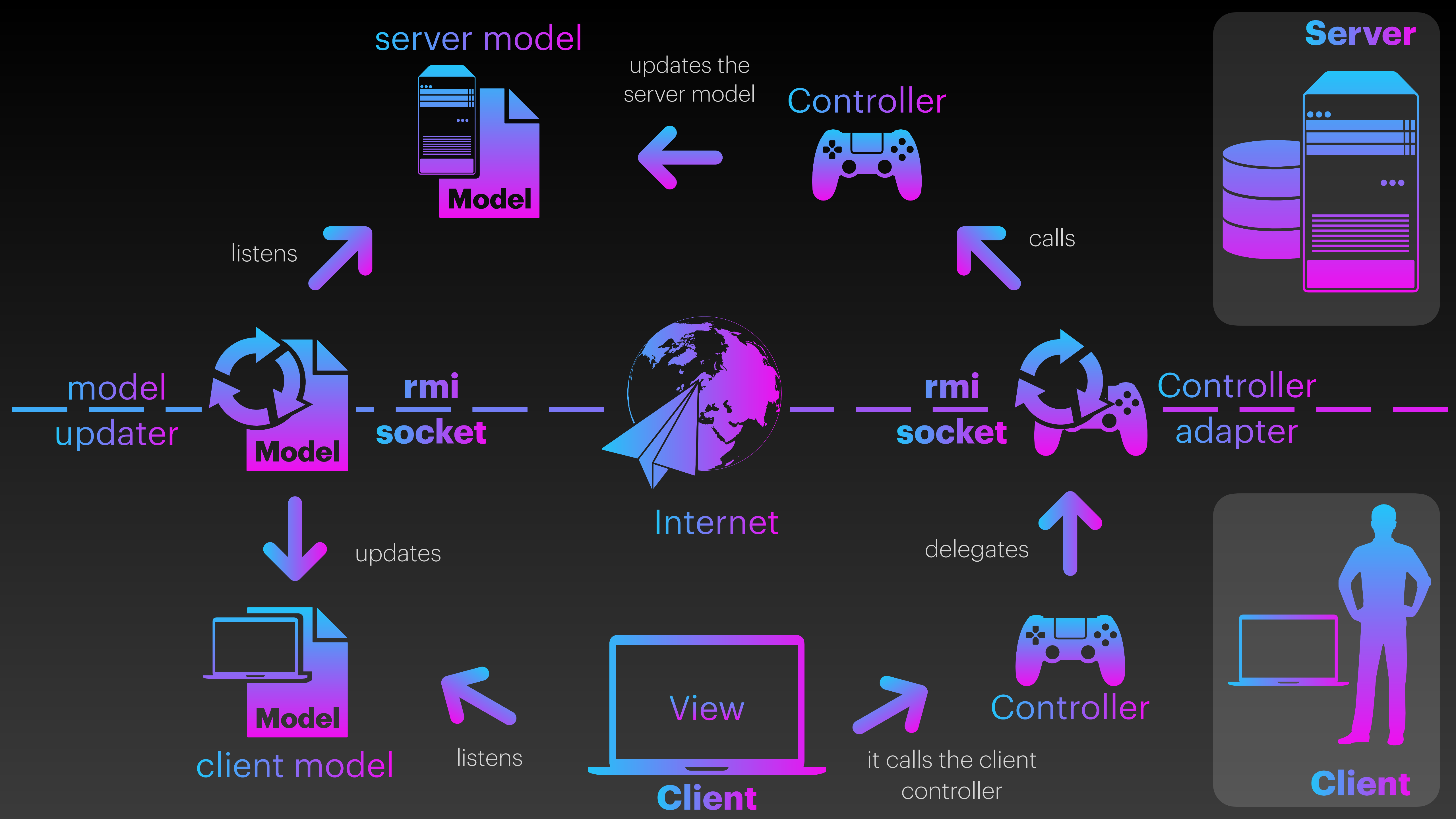


Ingegneria del Software



POLITECNICO
MILANO 1863

Francesco Ferlin - Francesco Guarda - Giacomo Orsenigo - Giovanni Sartorato



<<interface>>

Provider

T

get() : T

registerObserver()
unregisterObserver()

enables only **read**



<<interface>>

Property

T

set(val T) : void

enables **read** and **write**



Serializable Property

get() : T

set(val T) : void

registerObserver()
unregisterObserver()

class that implements
Property, that enables to
instantiate observable
objects (widely used in
client and **server** model)

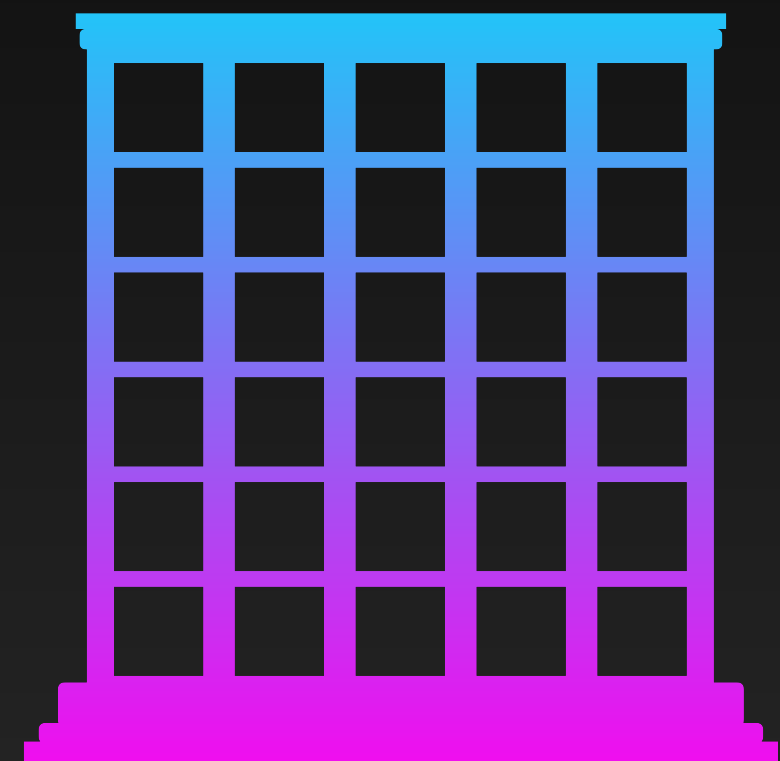
<<interface>>

Shelfie View

tile(r int, c int) : Provider<Tile>

Read-only object which
represent the shelfie of a
player in an in-progress game
therefore using **Provider**

Shelfie



Shelfie

shelfie[][] : Property<Tile>

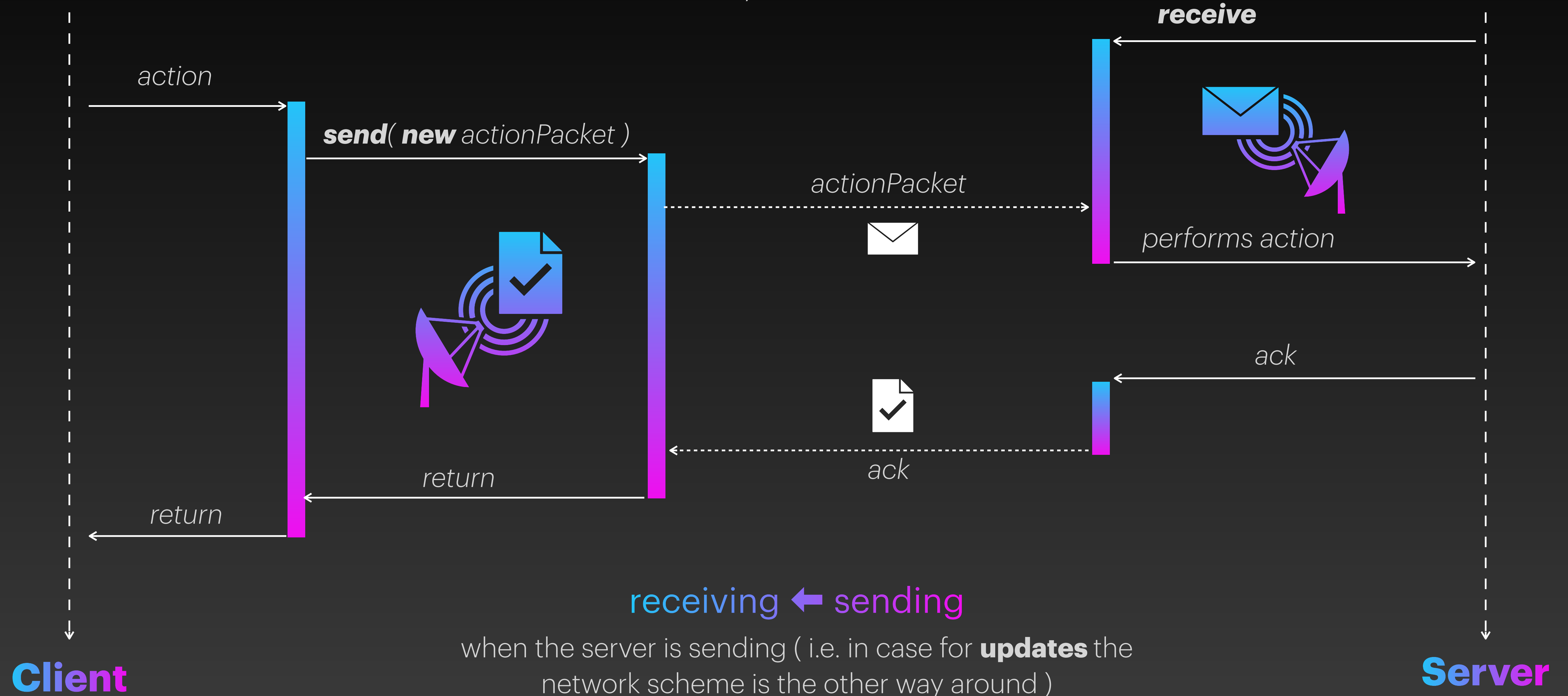
tile(r int, c int) : Property<Tile>

modifiable implementation of
ShelfieView therefore using **Property**

socket network communication

sending → receiving

using RMI most of the low level networking is abstracted
and resolved by the remote invocation



receiving ← sending

when the server is sending (i.e. in case for **updates** the
network scheme is the other way around)

makeMove() - socket sequence diagram

