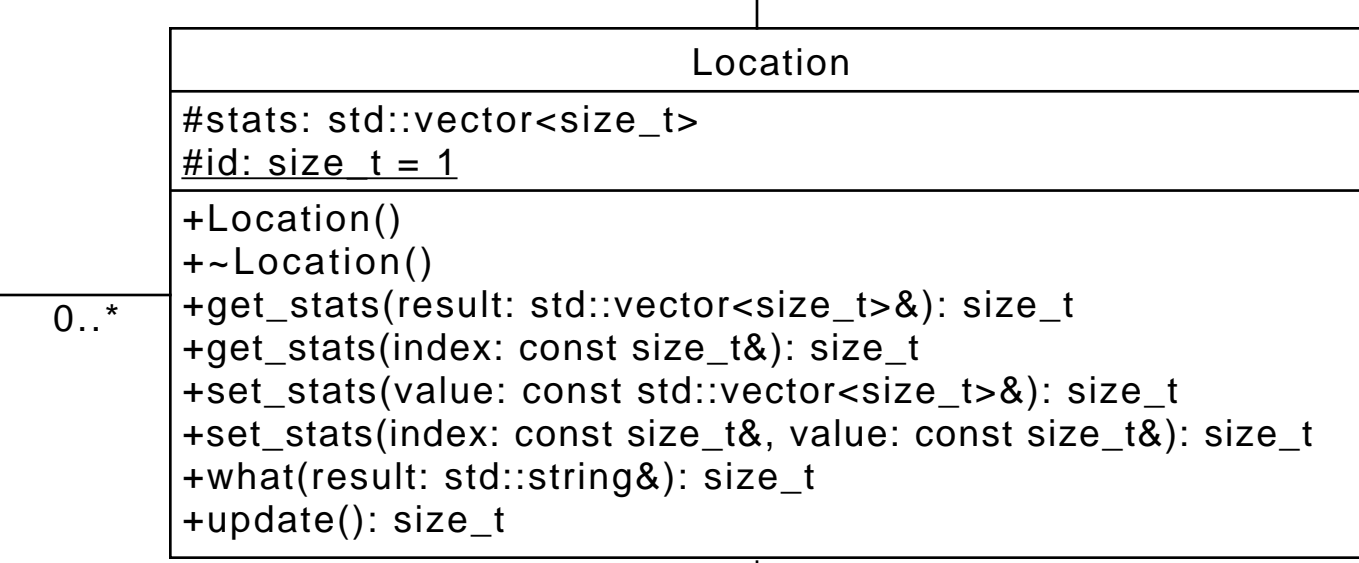
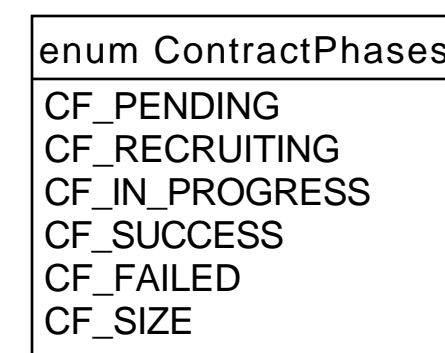


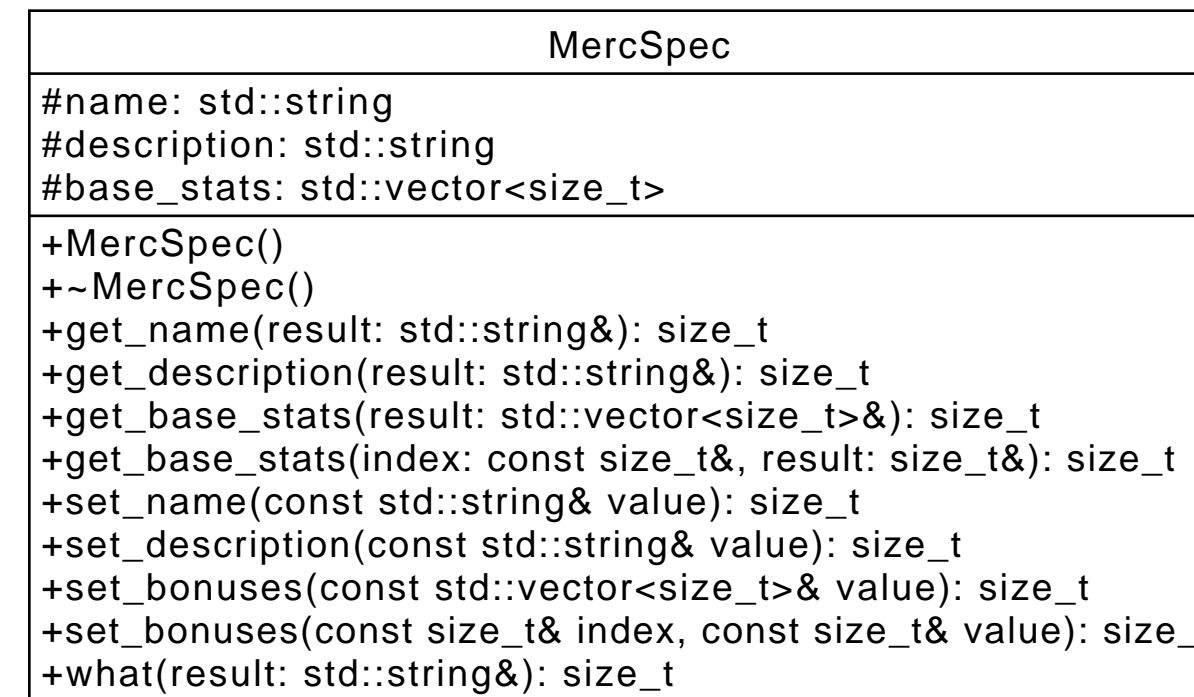
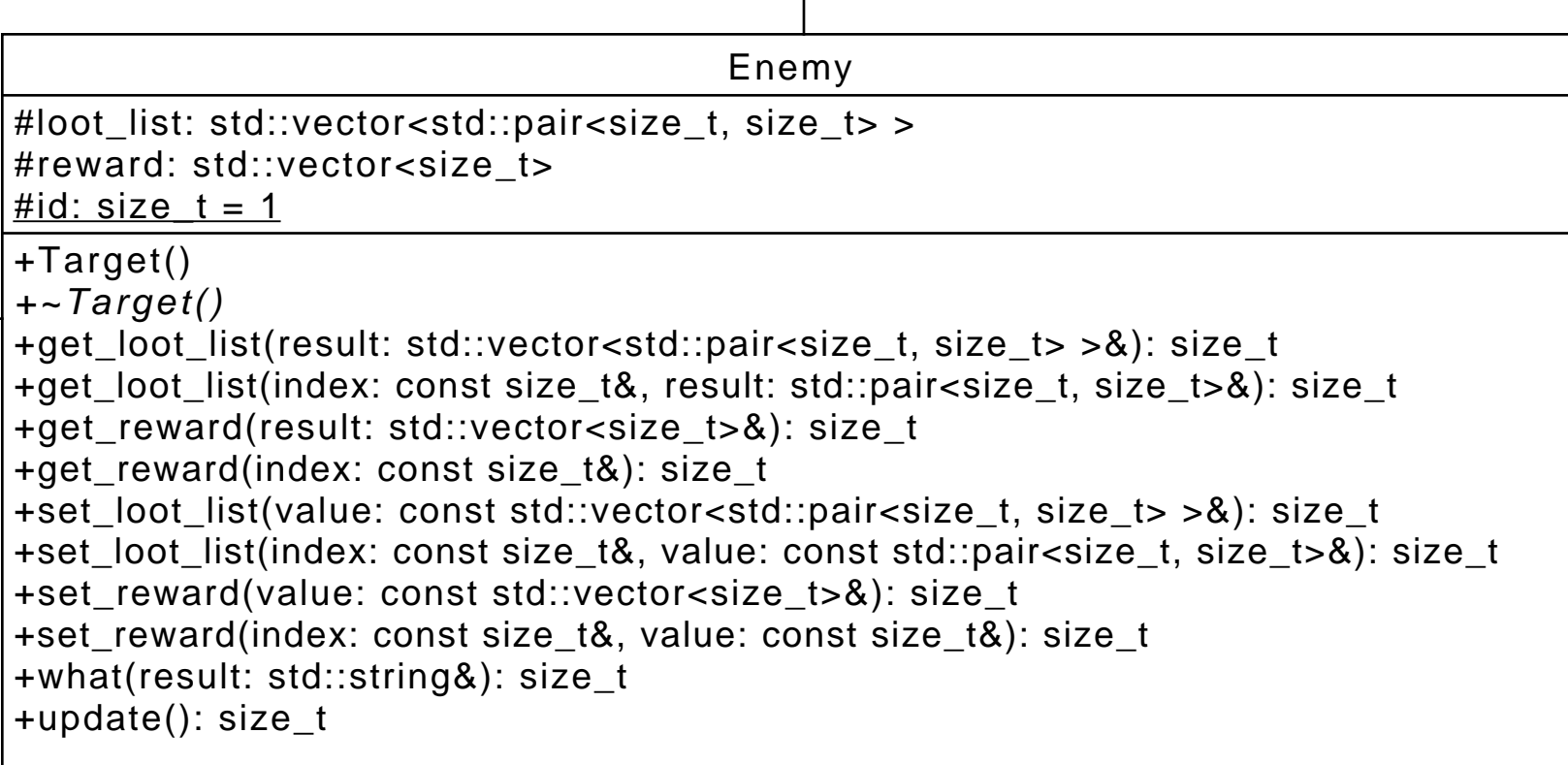
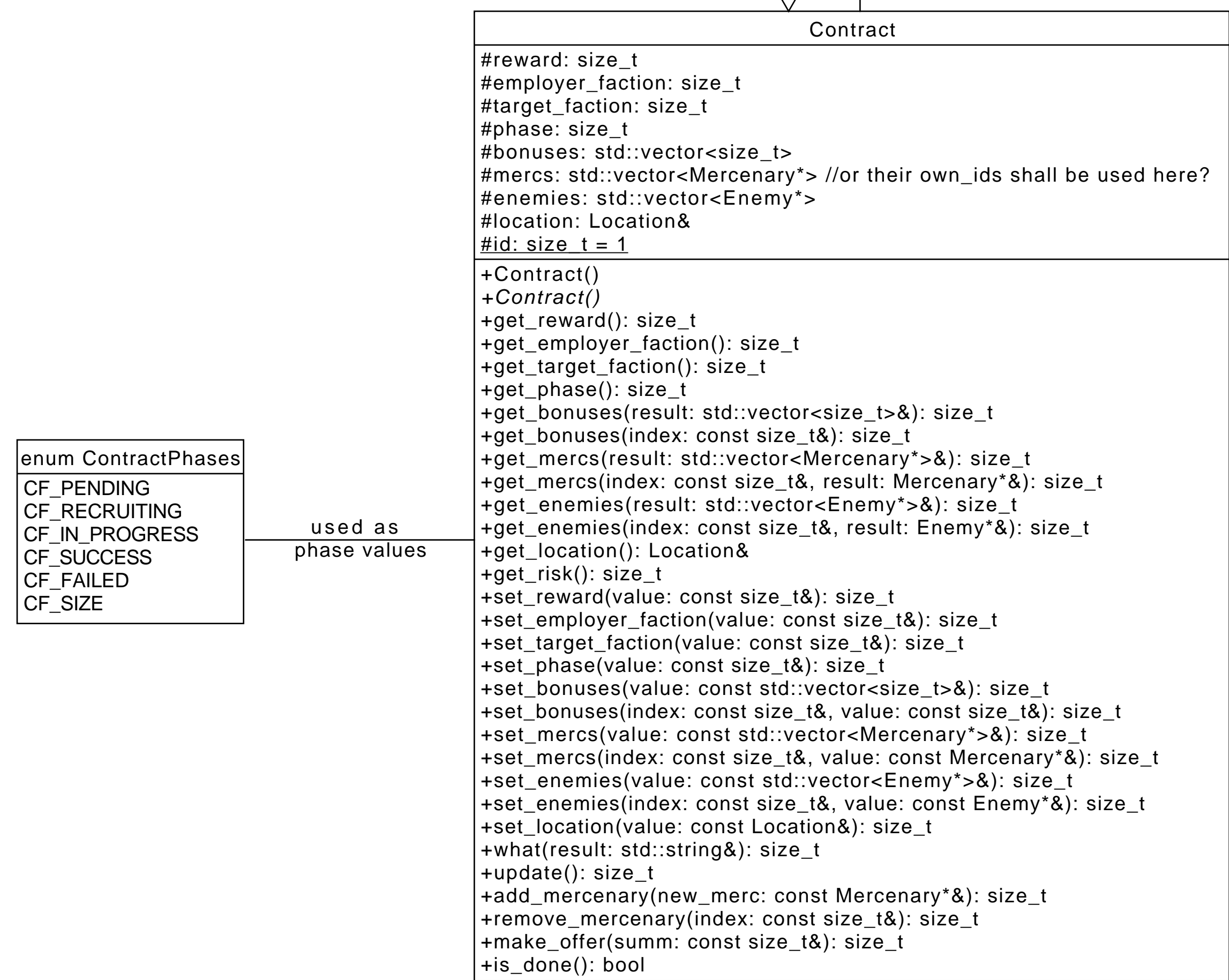
Actual questions:  
-Game mode: real-time or turn-based?  
-Engine for visualization? (proposal: Irrlicht)  
-packages for data storage? (proposals: sql database (via sqlite framework), json file(s) (via jsoncpp framework))



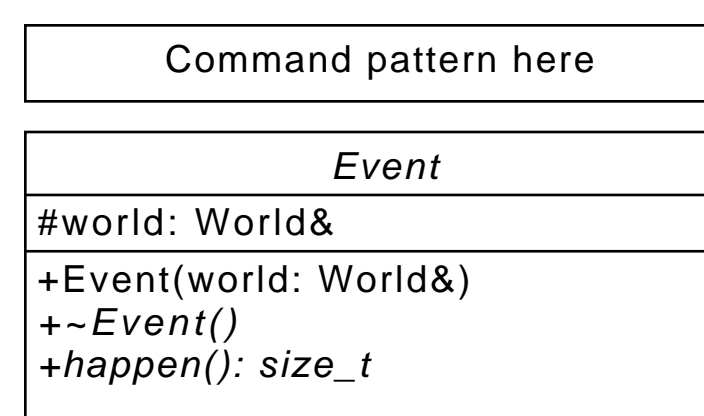
Type Object pattern



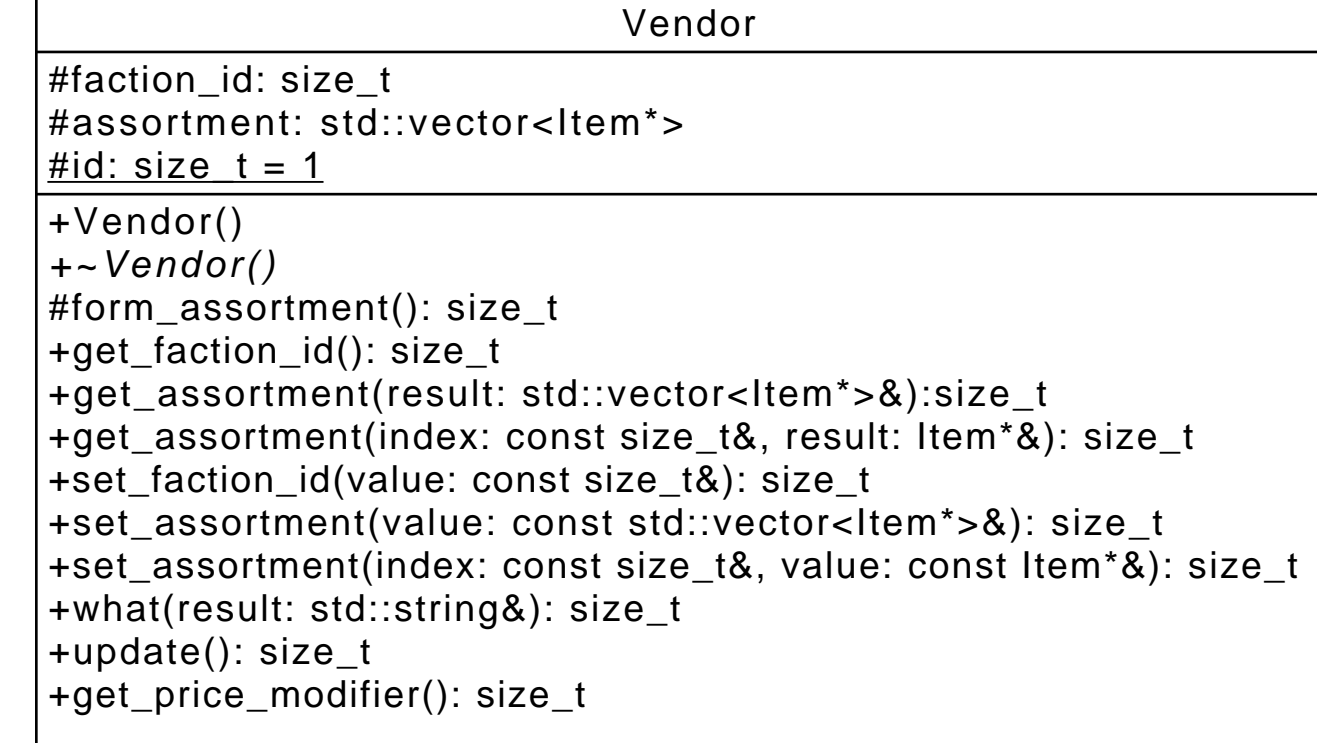
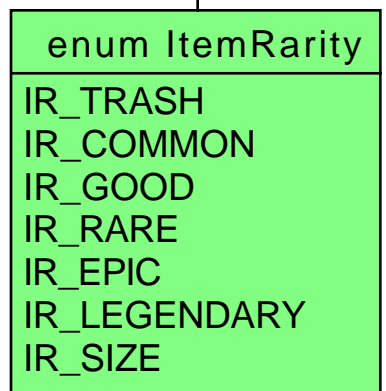
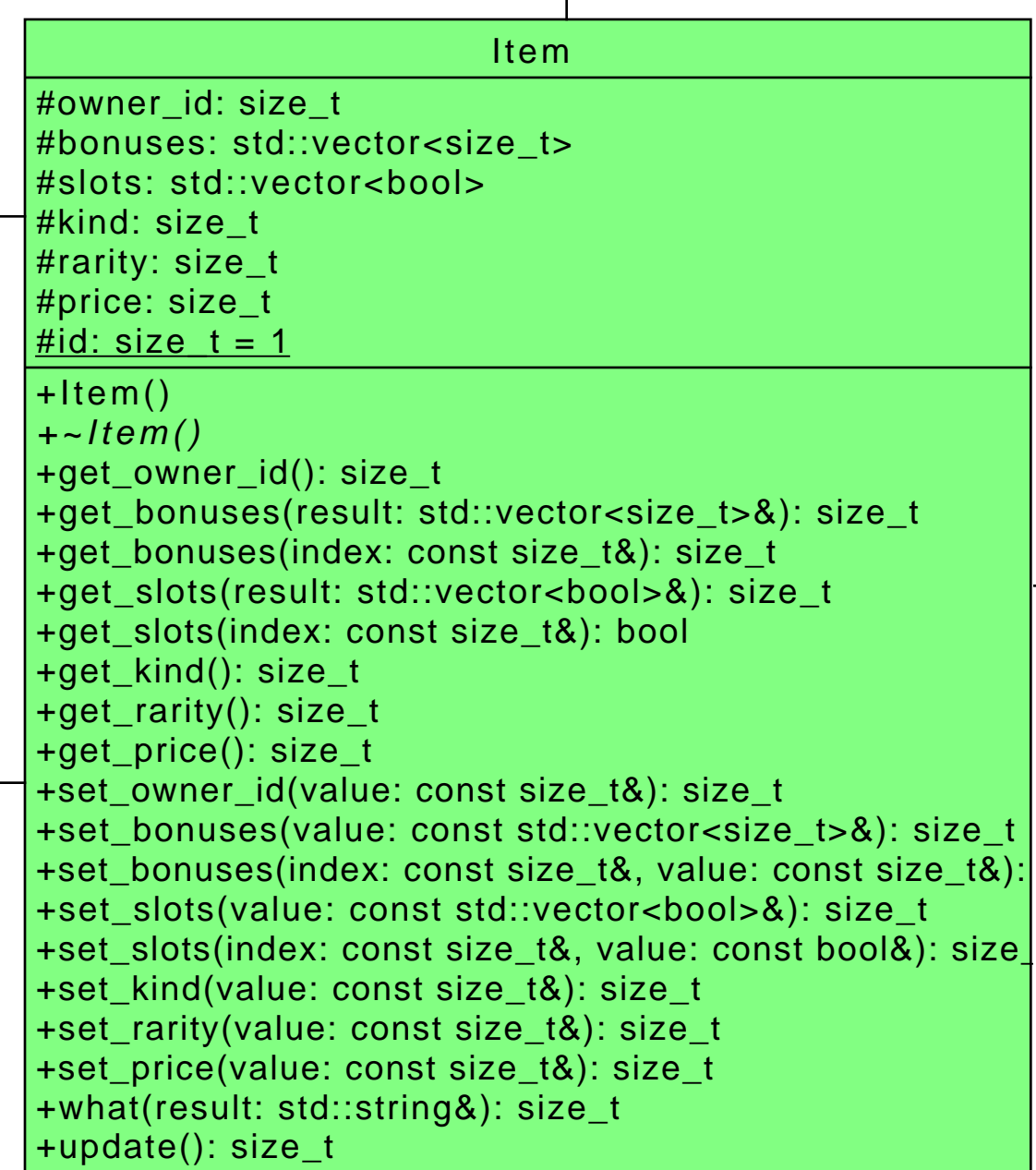
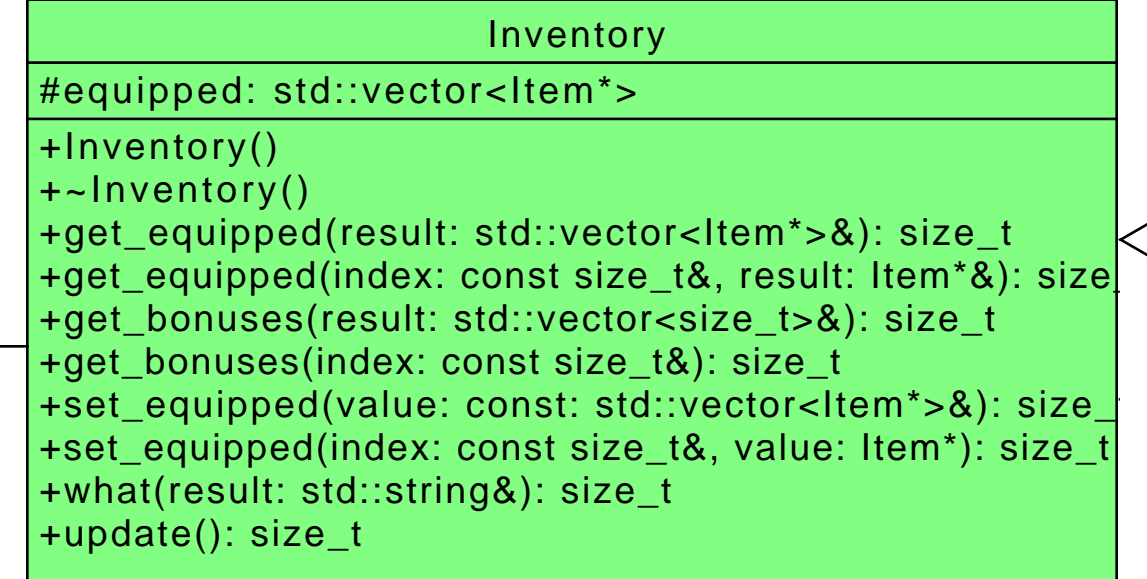
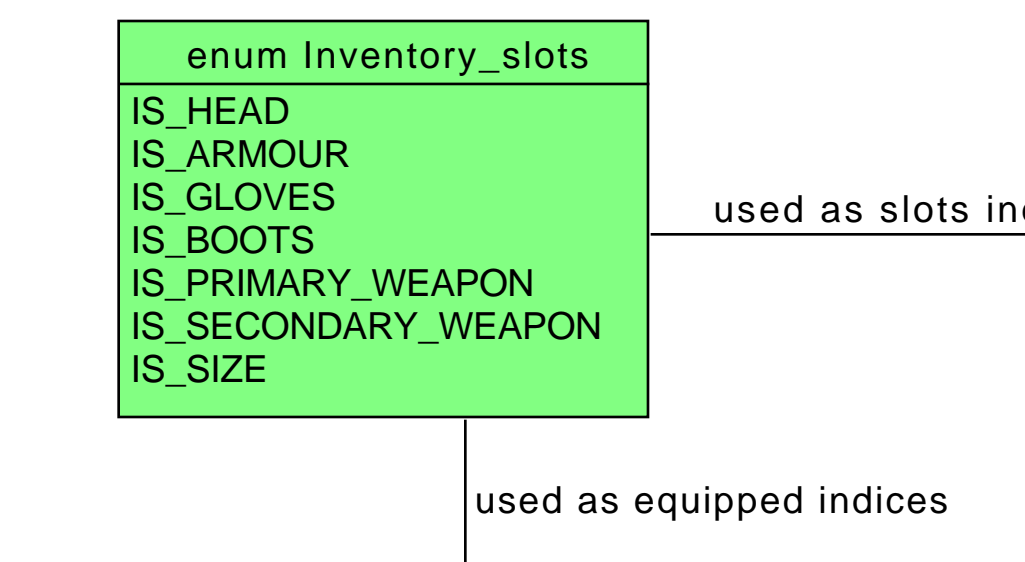
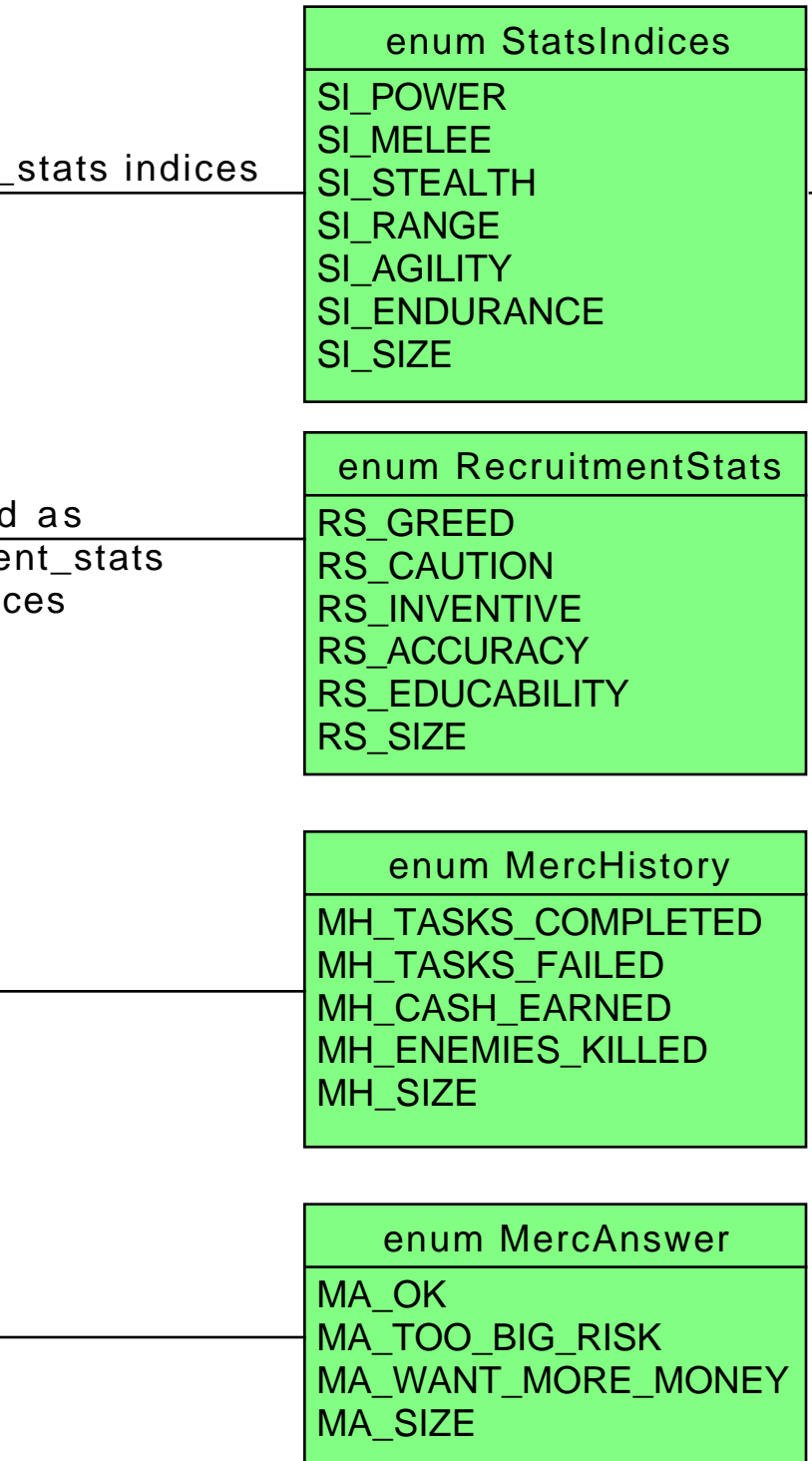
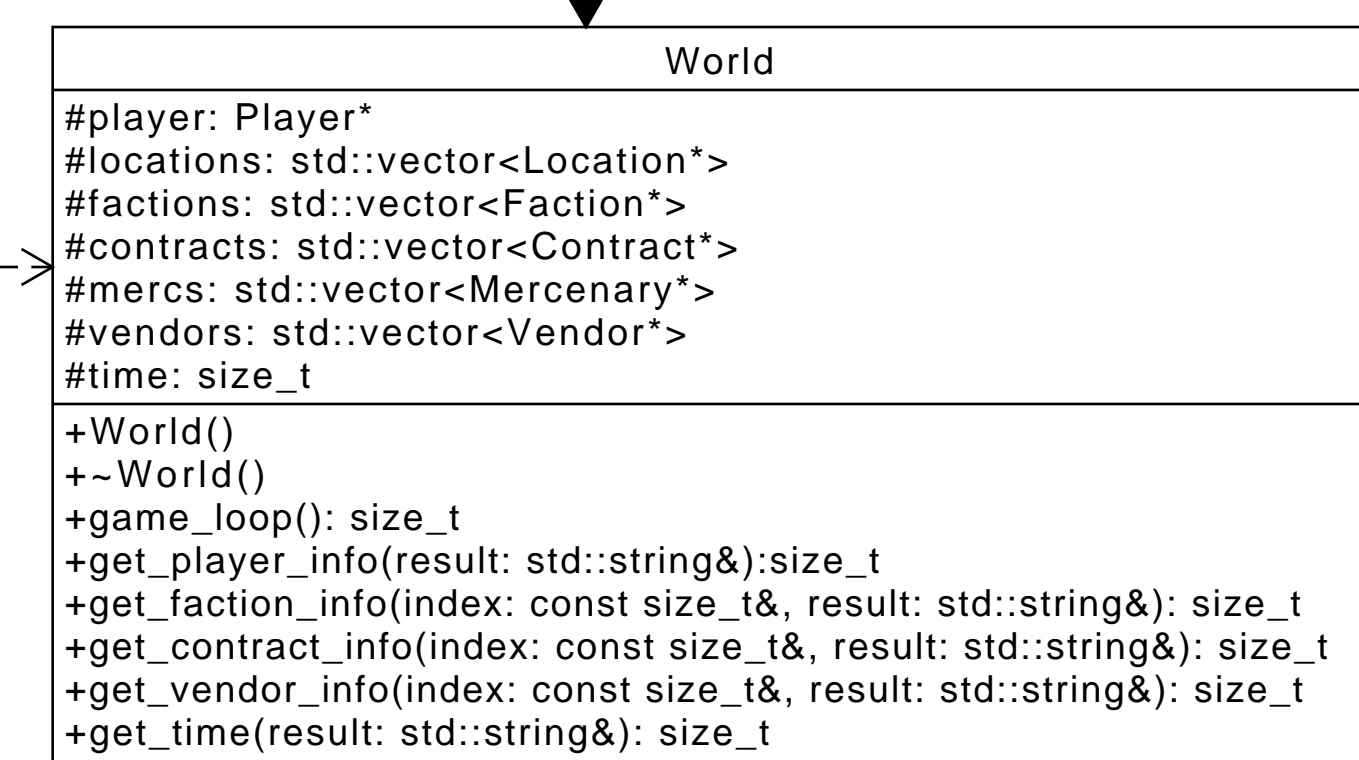
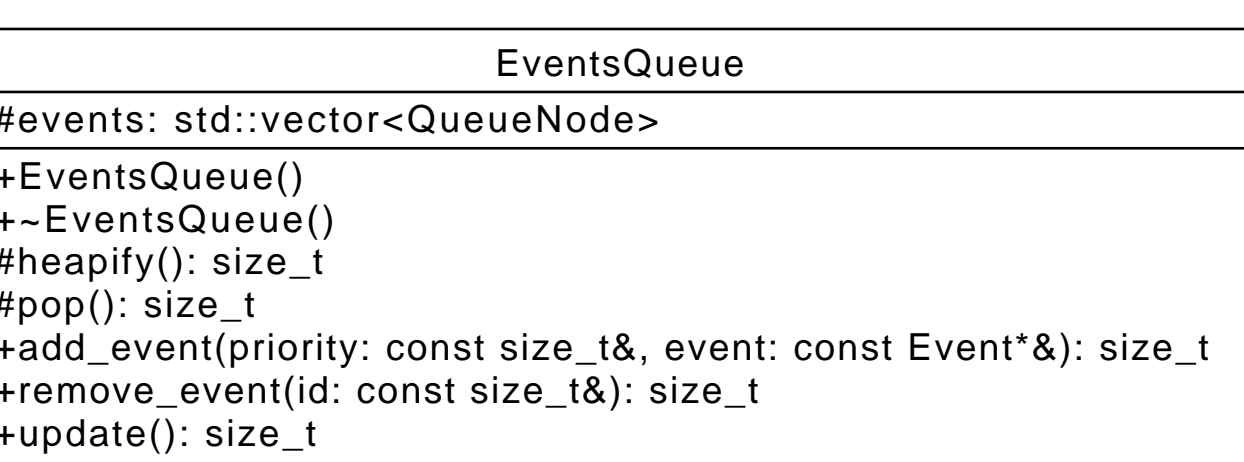
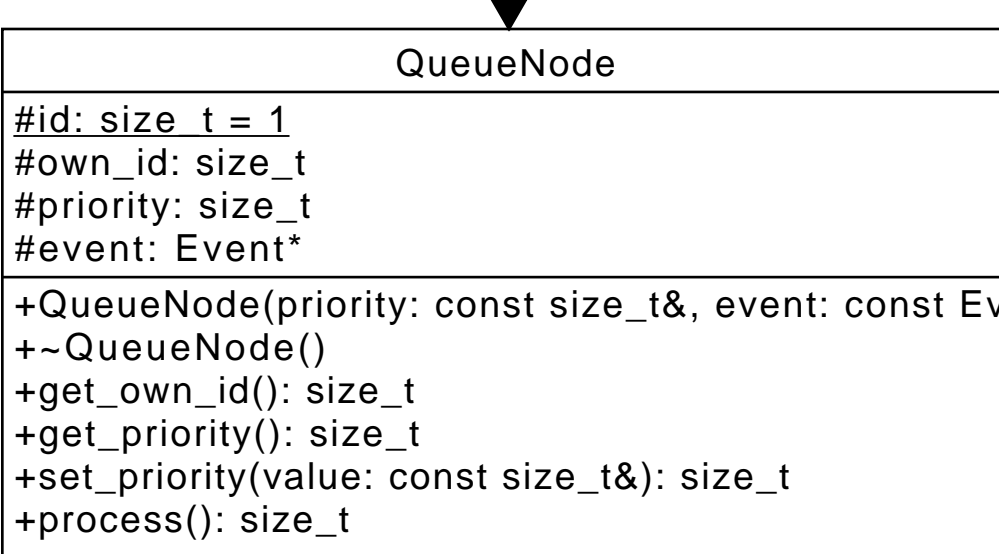
used as phase values



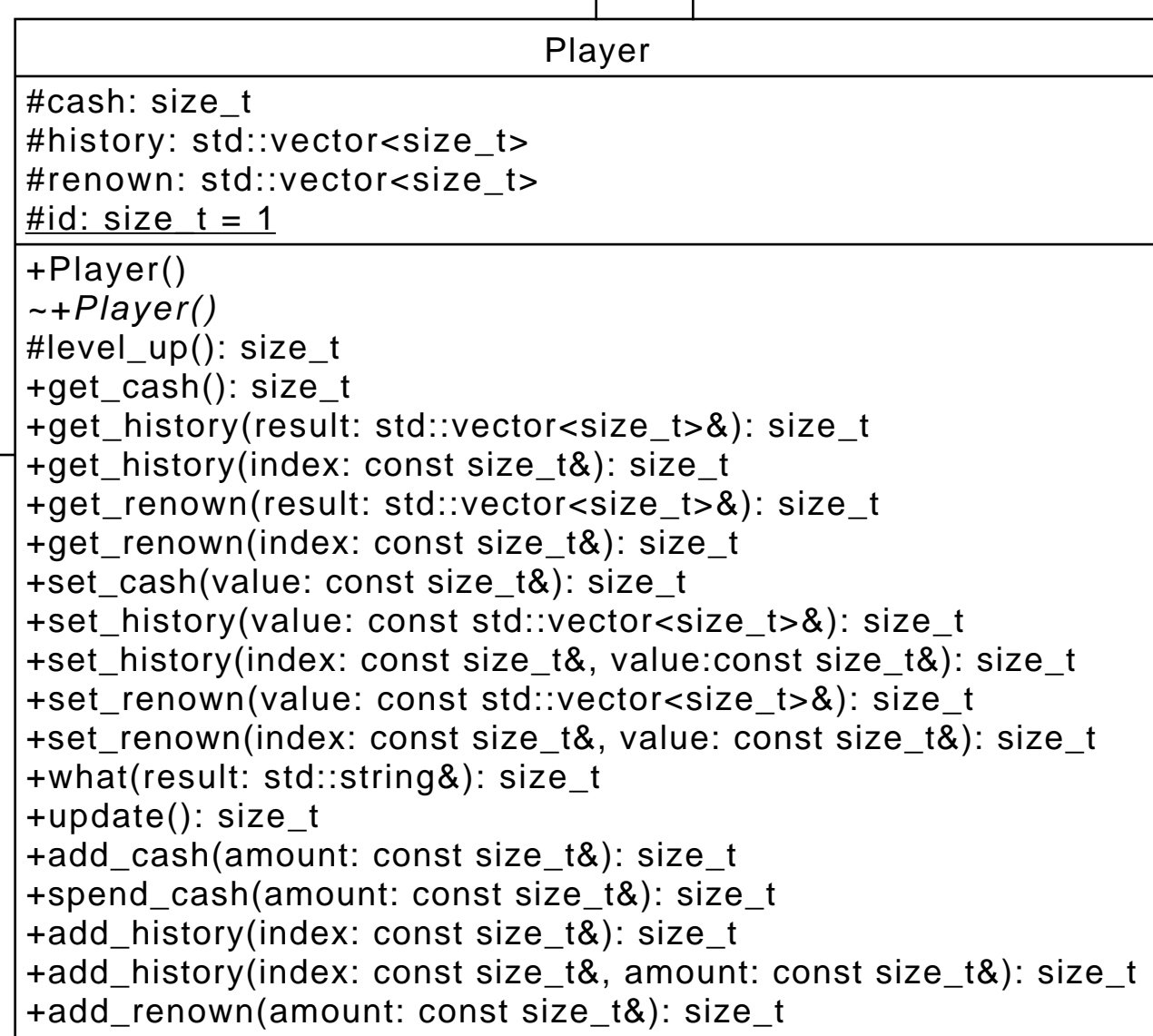
Not the best decision, other ways to provide Event's derivatives with access to World's protected fields?



Derived classes hierarchy here to realise various events, including user input handling.



buys and sells items



offers contracts to  
takes and completes contracts for

update() method pattern