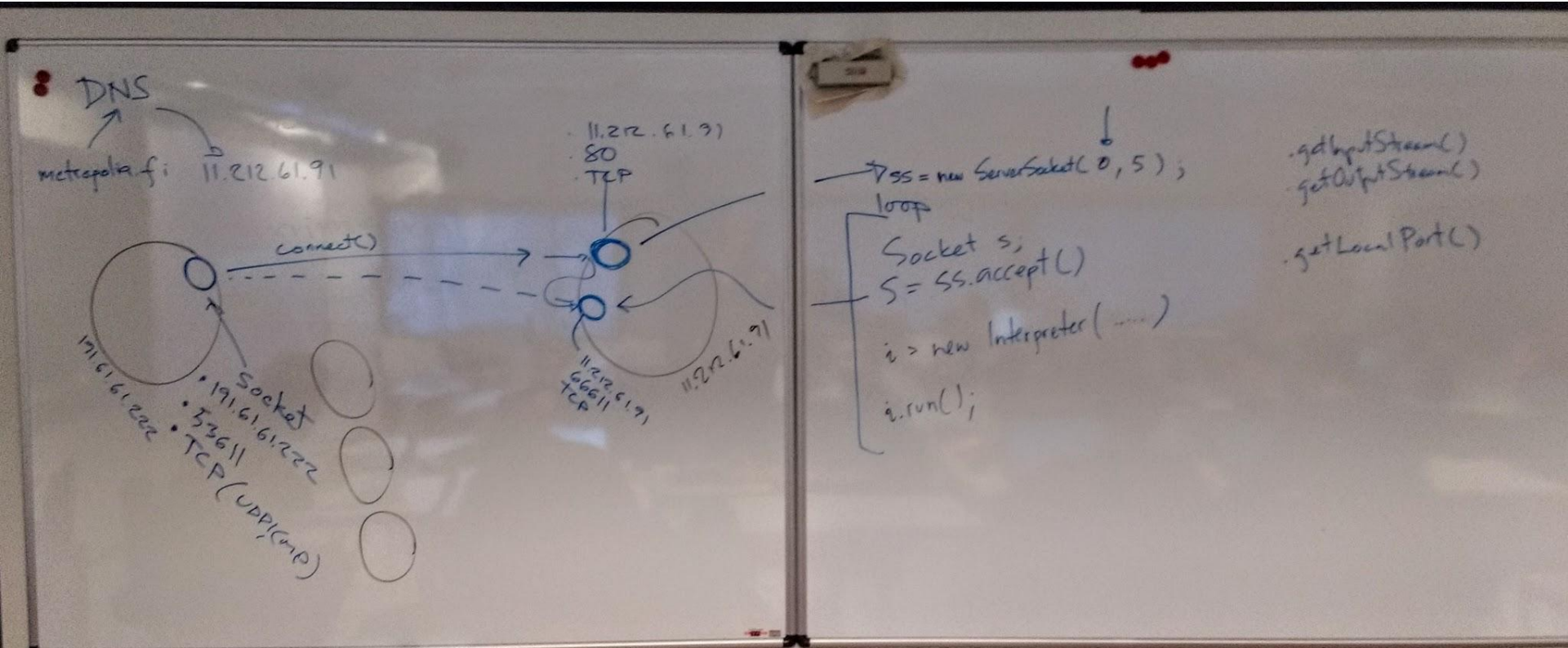
On sockets and threads

Sockets and getting connected



Sockets and threads in simple chat server

Make your Interpreter Runnable - it already has run() method

Create a new constructor for Interpreter - to be able to pass the socket to Interpreter, you can get input and output streams from the socket object

Make up the *name* instance variable in Interpreter from the socket information (IP address and port, for example)

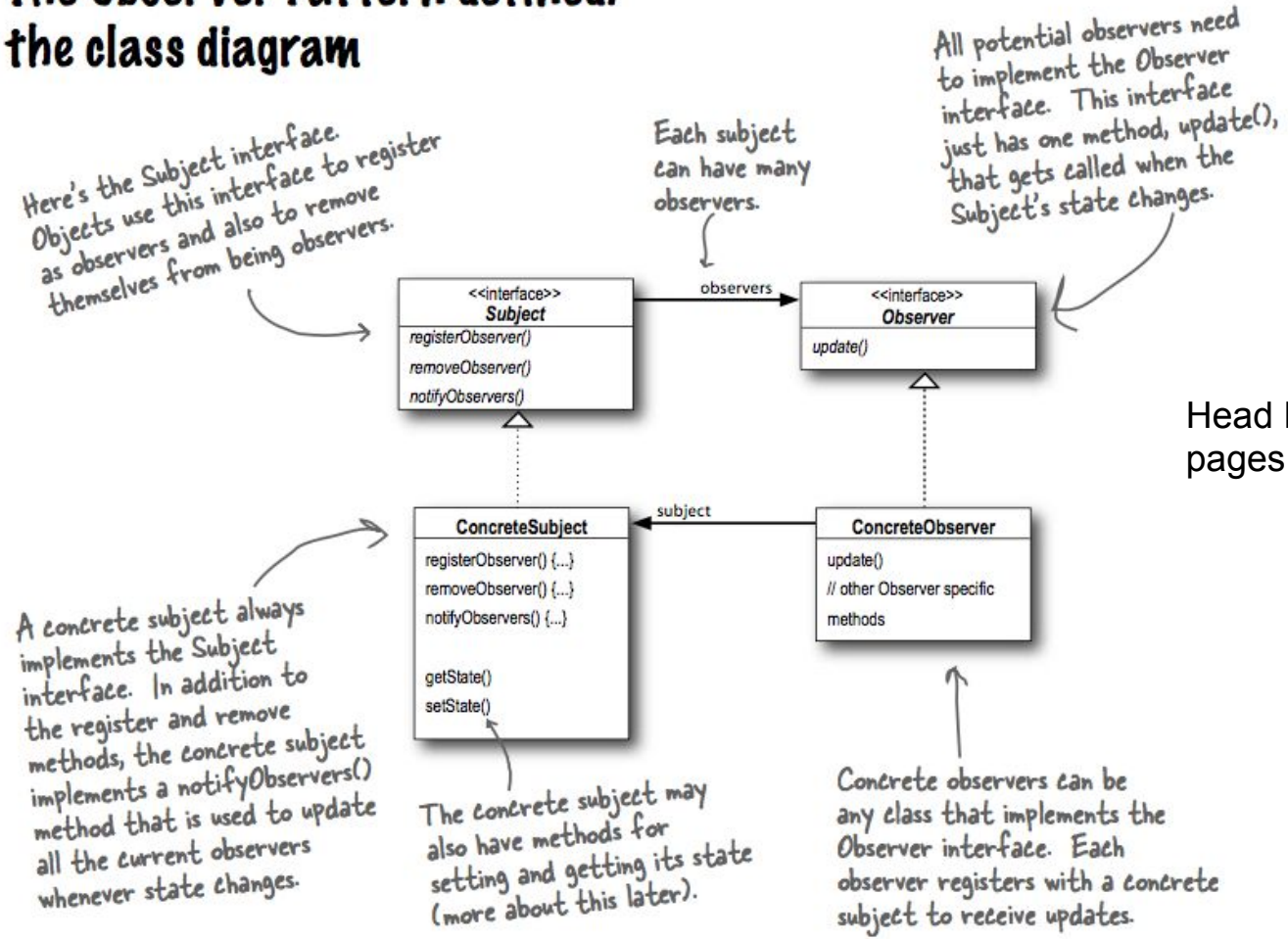
Observer pattern in simple chat server

Interpreter should implement HistoryObserver interface and register as observer when it starts and deregister when it is about to exit

History should implement methods to register() and deregister() an HistoryObserver

In History, make sure to call an update method defined in HistoryObserver interface to keep History observers updated

The Observer Pattern defined: the class diagram



Head First Design Patterns, pages 37-78

Reading list

On sockets:

https://docs.oracle.com/javase/tutorial/networking/sockets/index.html

On threads:

https://docs.oracle.com/javase/tutorial/essential/concurrency/procthread.html

https://docs.oracle.com/javase/tutorial/essential/concurrency/threads.html

On observer patterns:

Head First Design Patterns, pages 37-78