Assignment 1: Design

10/20/17

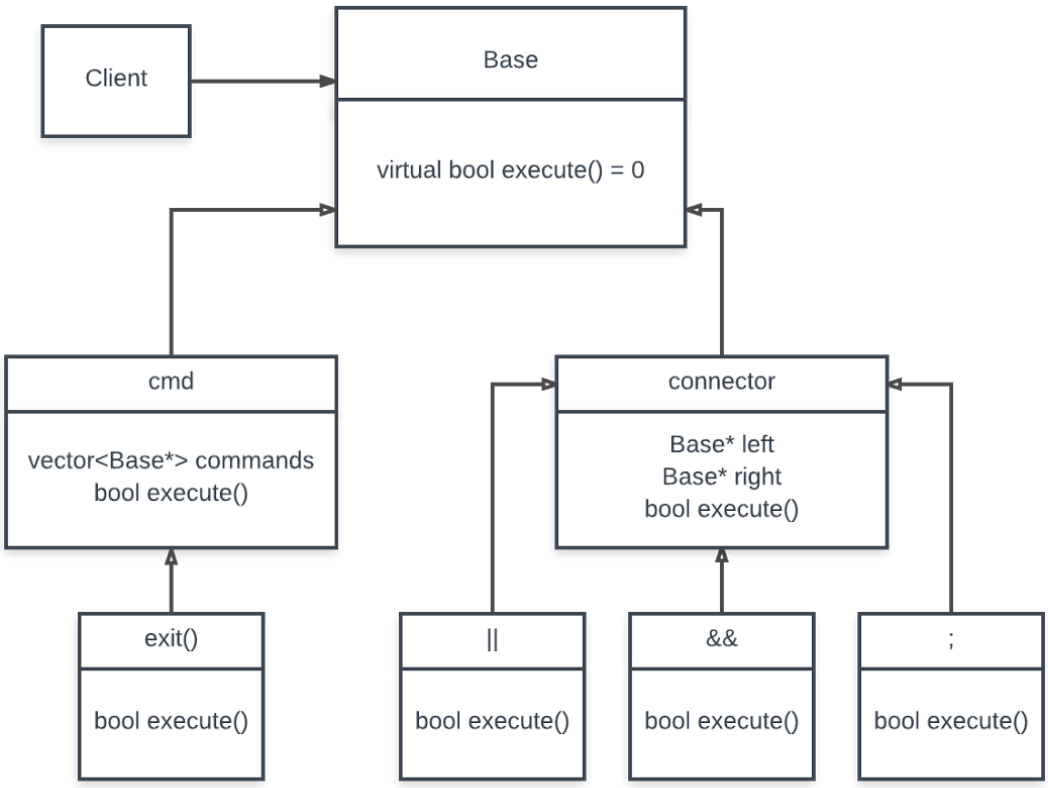
Fall Quarter 2017

Andrew To and Patrick Tumbucon

Introduction:

Rshell is a command line interface written in C++. It will execute bash-like commands with connectors (||, &&, ;). We will be using a composite design strategy where “cmd” and “connector” act as composite classes.

UML Diagram:



Classes/Class:

Base

This abstract class contains a pure virtual function, execute( ), which is implemented in all of our derived classes.

cmd
The cmd class will hold a vector filled with all the inputted commands and will execute them accordingly. This class is also a composite as it contains a derived class, exit( ).

exit
The exit class is primitive as it will inherit the functions of the composite, cmd, and that of the base class.

connector
Connector is the second composite of our design. It holds two Base pointers so that one may hold the value of the command before the connector, and the other will hold the command after the connector.

||
The || is a leaf of Connector as it contains the execute function from the Base. It will execute the second pointer, rhs, if lhs does not execute.

&&
The && is a leaf and will only execute if both lhs and rhs execute.

;
The ; is the third leaf of Connector and executes the lhs and rhs.

Coding Strategy:

We will primarily do pair programming from the ground up to ensure that we're on the same page. When pair programming is not possible, we can divide the work between the two composite classes and their derived classes.

Roadblocks:

It is likely that we'll have compatibility issues with our classes if we are not careful enough. This can be prevented by starting from the Base class and pair programming.