

Part 1

Acme Works just contracted "Pat's Contracted Software" to design and build a set of coffee machines for their work place. There are mechanical designers taking care of all the mechanical parts of the machines. You are responsible for the machine's controlling software.

Acme's requirements are very simple. They want to sell coffee, with or without cream powder and sugar, for \$0.35. The front panel will have 4 buttons, for Black, Cream, Sugar, Cream&Sugar.; There will also be a Coin Return button. The machine will take (American) coins only, and give change.

Any other displays are up to you to decide on and design. There will be dispensers that can dispense a fixed amount of: cup, hot water, coffee powder, cream powder, sugar. You and the member of marketing who will be visiting your team can negotiate any other requirements you feel you need resolved.

Pat has heard so much about this "object oriented" software stuff and all its benefits, that he wants the controller designed using OO techniques. It can be implemented in C++, Java, Smalltalk, or even assembler or CLOS for all he cares. However, this is a simple problem, so he expects the job done fast. Since this is your team's first problem, he'll give you two hours to sort out your requirements, design process, argue, document the design and sketch out the code. He wants to see at least one interaction diagram or collaboration diagram showing how the machine operates. If you have time, draw a second one showing how the machine reacts when given insufficient money, or when it runs out of sugar.

Part 2

Acme Works' bosses are happy with their coffee machine, although they grumble at how long it took to get them. They want a small alteration made to the machines. Since the change is small, and they heard Pat brag about this great OO implementation stuff, they expect the change to be quick and cheap. Pat agrees.

The alteration is simply that the machine should also serve Soup, for \$0.25. The mechanical designers said this is no problem, and are fitting the machine with one more button for soup, and one more powder dispenser.

Pat expects you to document the design alteration, again using at least an interaction or collaboration diagram showing the machine in operation, and have the new machine coded in 30 minutes, since it is such a small change. If you need more time than this, you will have to argue for it. Spend some time, after you make the modification, in discussing how using OO techniques did or did not help you minimize the changes to your system.

Part 3

Fu Man Chu Moustache Grooming Emporium likes what Acme got for machines, and so they just contracted "Pat's Contracted Software House" to design and build a set of coffee machines just like Acme's. The only thing they want different is that their vending machines can read the company

badges to directly charge the cost of the purchases to the employee's account.

The marketing staff have located and bought the right kind of card reader and another department has provided a network attached device for communicating with the network server to keep track of the purchases. All you have to do is change the design so that the machine accepts badges or cash, and sends any badge-purchases to the server. The badge is simply encoded with the employee number.

Pat figures this change will be simple, so he expects you to alter the machine, document the changes in your design and sketch out the changes to the code in 30 minutes.

Part 4 and Finale

Pat sees a market in these coffee machines, so he want the next set to allow his clients to compete with Starbucks. He wants it to have espresso powder and a milk steamer. The mechanical department says this is no trouble. The new machine will have 6 powder dispensers (initially for espresso, coffee, milk powder, sugar, soup, tea), as well as the milk steamer.

Pat wants to be able to run a "drink of the week", to be espresso, or cappucino, or latte, or latte mocha, or double espresso. Any given one of those will have a matching price, for example, \$0.75 for an espresso, \$1.50 for a latte or cappucino, \$1.75 for a latte mocha, \$1.00 for a double espresso. The people designing the front panel are making changes, so that the buyer can select from the available drinks on a list on a programmable LCD touch panel.

Pat figures this is a big change, but maybe your object-oriented design lets you do it faster than otherwise. So he gives you one hour to make the change, document the design, sketch out the behavior of the new system, and sketch out the code packages. The most important document is the interaction/collaboration diagram showing the system in operation. If you have time, show the system reacting to being out of an ingredient.

Finale: Discuss your designs, and their evolution. Reflect on the way you worked as a group, and on how your design process evolved over the problems. Be prepared to post your interaction diagrams.