CarSharing application

Three entity classes are used in a collaboration - CarSharer, Journey and Address. Each of these classes will be implemented by a (.java) source file. these classes are used across a number of use cases and are grouped together into a CarSharing component as Java (.class) files. Here we are just dealing with the source files. There are two other classes MCSUserInterface and MCSControl. Each of these will eb implemented by a (.java) file. The MCSControl component has a dependency on the CarSharing component and on the MCCSUserInterface Component.

A) Draw a component diagram showing the source ocde dependencies. The > .class file are grouped together into two Java archive (.jar) > files. The MCSControl.class component will need to read a > configuration file (MCS.ini) and display a help file (MCS.hlp) > when required. The MCSControl (.class) file also has dependencies > on the MCSUserInterface (.java) file and the CarSharing (.jar) > components.

> {width="4.947916666666667in" height="3.25in"}

ref: U0812 @Peter LO 2010

> B) Re-draw the component diagram to show the run-time component dependencies.

{width="4.458333333333333in" height="3.7395833333333335in"}

> ref: U0812 @Peter LO 2010
>
> C) Draw a deployment diagram given that the nodes are three client PCs, a server and a printer. The communications protocol between the clients and server is TCP/IP; and between the server and the printer is a standard parallel printer protocol. The MCS.jar user interface and the control objects will run on the clients. The CarSharing.jar will run on the server. Server will also provide initialisation file MCS.ini and an help file named MCS.help.

{width="5.916666666666667in" height="3.5833333333333335in"}

> ref: U0812 @Peter LO 2010