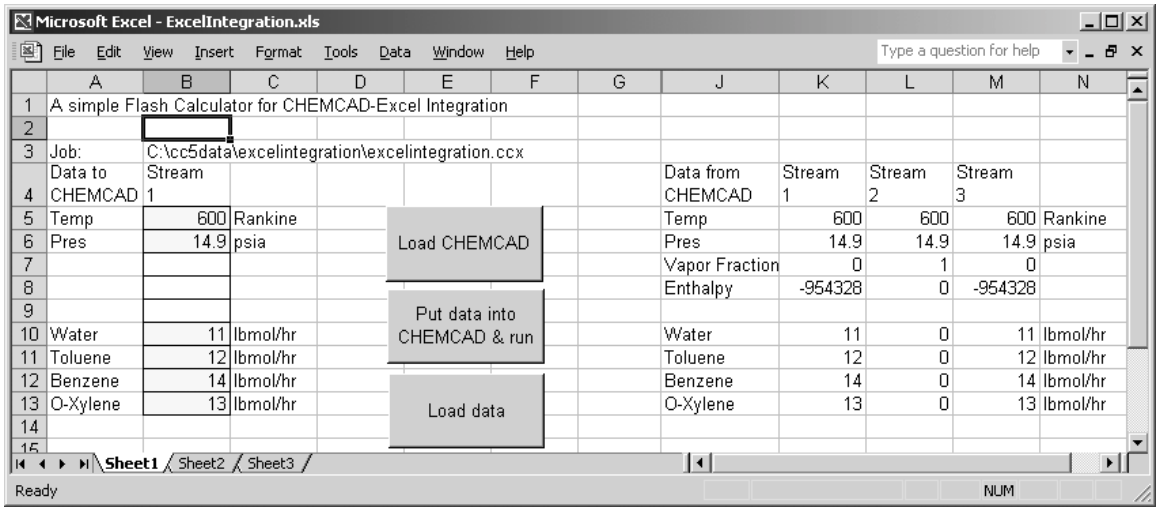
Integration with Excel: The ExcelIntegration Example

Aaron Herrick

The VBServer interface allows users to Start CHEMCAD, load a simulation case, transfer information back and forth from another application, run the simulation, etc in an automated fashion. This approach enables any Visual Basic application (Such as Microsoft Excel) to become a gateway to CHEMCAD. Since most if not all plant information systems already interface to Excel, Excel becomes a bridge between plant simulations in CHEMCAD and the plant data.

Using ExcelIntegration

To use the Excel integration example, simply start the ExcelIntegration.xls worksheet. If Excel asks if you want to enable or disable macros, choose "Enable", otherwise the example won't work.



The ExcelIntegration workbook

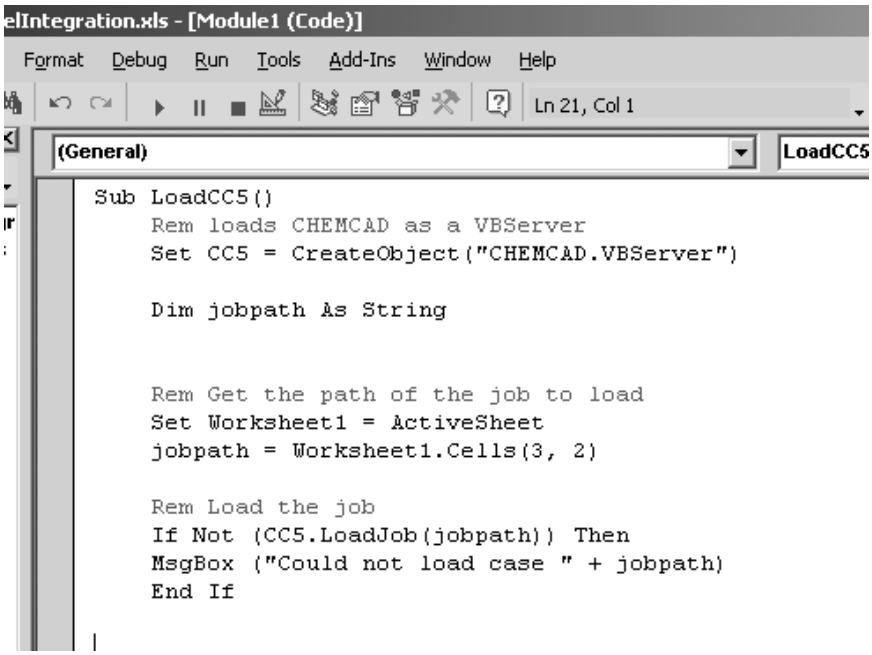
This workbook is set up as the interface to a very simple flowsheet consisting of a single flash unit. The filename and path in cell "B3" denotes the path to the simulation Excel will be using. The rest of the data in the B column represents stream 1 in the flowsheet.

There are three buttons in the center of the workbook: **Load CHEMCAD**, **Put data into CHEMCAD & run**, and **Load data**. The first button, **Load CHEMCAD**, loads CHEMCAD into memory and opens the specified job in CHEMCAD. The second button transfers stream data from column B into CHEMCAD and runs the simulation. The third button transfers information out of CHEMCAD into columns K,L, and M.

To use the workbook, first press **Load CHEMCAD**. Next, enter temperature, pressure, and flowrate information into column B, and press the **Put data…** button. Finally, press the **Load data** button to transfer data out of CHEMCAD back into the workbook. It may help to open CHEMCAD yourself and load this simulation so you can see what is happening.

Understanding ExcelIntegration

The buttons shown of the workbook are used to call 3 visual basic macros: **LOADCC5()**, **Data_to_CC5()**, and **Data_From_CC5()**. These macro's define all the communication between CHEMCAD and the ExcelIntegration workbook. To examine these macros in detail, right-click on the **LOAD CHEMCAD** button, choose "Assign Macro…" from the menu, and then press the **Edit** button on the following screen. You should see the Visual Basic editor load, and you'll see the LoadCC5 Macro displayed:



The LoadCC5 macro

The LoadCC5 macro is very simple, there are 3 important lines of code in it. The first important line is :

```
Set CC5 = CreateObject("CHEMCAD.VBServer")
```

This line loads CHEMCAD and associates it with the global variable "CC5". In the next important line, a variable "jobpath" is set equal to a cell in the workbook:

```
jobpath = Worksheet1.Cells(3, 2)
```

Finally we use an If/then statement to simultaneously load the specified job and check for errors:

```
If Not (CC5.LoadJob(jobpath)) Then
```

So, we can see how the macro's in the Excel workbook are accessing CHEMCAD via an object variable named CC5. The syntax and listings of available visual Basic methods is defined in the document "CC5CLASS.doc" which is stored in the CHEMCAD program directory (C:\CC5, typically) for easy reference. The next two macros are a little more involved because they are actually transferring data back and forth between CHEMCAD and Excel, but they use the same principles as LoadCC5 to do what is necessary.

Clearly, the ExcelIntegration example illustrates the ability to transfer data back and forth from CHEMCAD to Excel and vice versa. These methods are being employed daily to create plant optimization systems and further integrate CHEMCAD simulations into the plant. If you have any questions, please email me at aaronh@chemstations.net