Q Now and again a question pops up on the Bentley discussion groups about text files, and how to write data from MicroStation to a formatted file. The `TextFileWriter` example shows you how.

A Suppose that you have a MicroStation design file that contains some tag data, and you want to export the data for analysis in your favourite spreadsheet. You decide to export data from tagged elements to a comma-separated-format (CSV) file, a format that many office applications can read. I've included a design file with a defined Tag Set in the ZIP archive where you found this PDF file.

The Tag Set is named `LA_Solutions` and contains three Tag definitions …

Tag Name	Data Type	Default Value
`Name`	`Character`	`Box N`
`Value`	`Real`	`10.0`
`Quantity`	`Integer`	`1`

The method used to write tag data implements these steps …

1. Provide a name for the text file, using a Windows Common Dialog
2. Use a `FileSystemObject` from the Windows Scripting library to open the file
3. Create a `TextStream` object to write each line to the file
4. Scan the design file, using `ScanCriteria` set to find tagged elements
5. Extract each element's Tag Data and format into a line of text
6. Write the line to the `TextStream`

Once the design file is completely scanned, we have list of exported data in CSV format. I've provided a sample output file `tags.csv` in the ZIP archive. It looks like this …

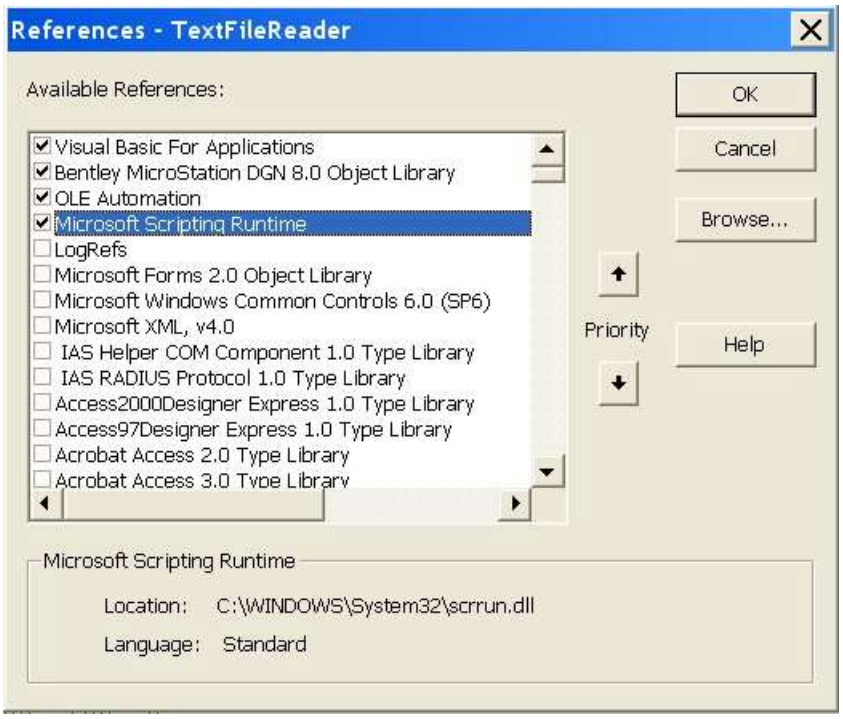
```
4,4.25,"Box 6"
1,30,"Box 7"
1,8.63,"Box 8"
3,49.99,"Box 9"
2,7.66,"Box 10"
1,9.99,"Box 1"
1,2.25,"Box 2"
1,25,"Box 3"
1,3.99,"Box 4"
2,5.75,"Box 5"
```

You can open a CSV file with a text editor, such as Windows Notepad or a more sophisticated product like TextPad. More usefully, you can open it with a spreadsheet or import it into a database.

The ZIP archive includes MVBA project file `TextFileWriter.mvba`. Unpack this archive into one of the folders in your `MS_VBASEARCHDIRECTORIES` configuration variable; normally, this includes the folder `C:\Program Files\Bentley\Workspace\Standards\vba`. Next, start MicroStation and load this project. You'll need to add a reference to the Windows Scripting library, which we use to create a `TextStream` object for writing to a text file. In the VBA editor, choose the Tools|References menu. Scroll down the list to find the library, and check the box on the left. Click the OK button to complete adding the reference.

Open the supplied file `tags.dgn`, or any other file that contains tagged elements. Run `TextFileWriter` from the IDE, starting at procedure `Main`, or from MicroStation using the keyin `vba run [TextFileWriter]modTextFileWriter.Main`. The code asks for a save file name, then scans the design file for tagged elements. For each tagged element it finds, it formats a line that is written to the named output file.

The VBA code doesn't just scan the active model: it also scans any referenced models. This illustrates how to iterate the reference collection, and to extract tag definitions from model references other than the active model.



The MicroStation VBA project `TextFileWriter.mvba` is ready for action and browsing, so you can see the code described above. You can download it from the Publications page of our web site. It's in a ZIP archive with a sample design file, unimaginatively named `tags.dgn`.

