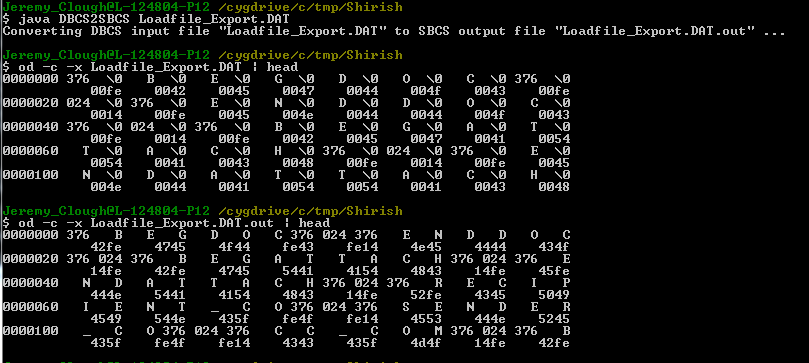
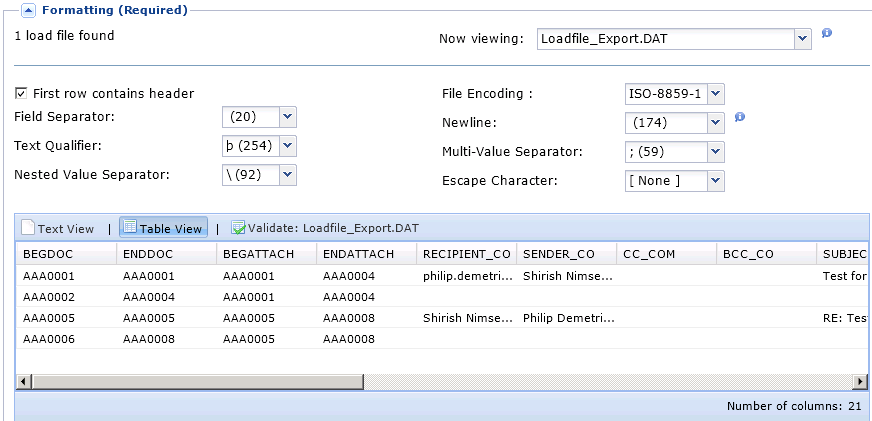
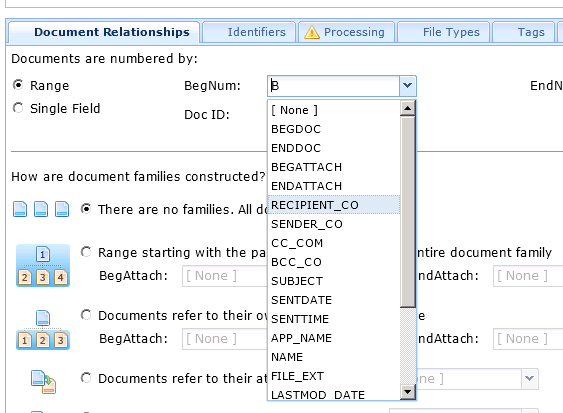
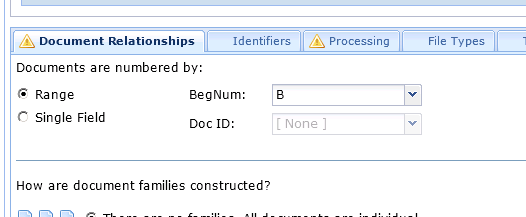
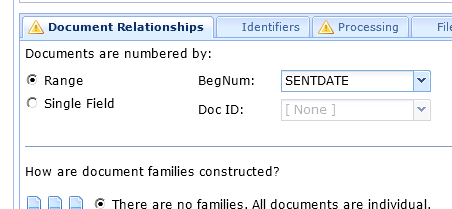
**DBCS2SBCS Example usage**  
You can run the Java program from a DOS or Cygwin shell.   
  
Here I am running it from a Cygwin shell, so that I can use the od command afterwards, to examine the file content:  
  
  
  
The ‘bad’ double-byte character set (DBCS) version of the .DAT file initially looks fine in the CW LFI UI:  
  
  
  
However, looks what happens when you select one of the fields using a combobox:  
  
  
  
So, these all look fine, but pick one of them and this is what you end up with:  
  
  
  
Compare this with a version of the .DAT file that has been converted to single-byte character set using DBCS2SBCS:  
  
  
  
  
**Conclusion**:  
CW LFI can only partially handle this particular double-byte character set (DBCS). My DBCS2SBCS conversion tools indicates that this particular DBCS format is “UTF-16BE”. According to <http://docs.oracle.com/javase/1.4.2/docs/api/java/nio/charset/Charset.html>, “UTF-16BE” is described as: “Sixteen-bit UCS Transformation Format, big-endian byte order”. This is supposed to be supported by Java, so perhaps the Clearwell issues are Javascript related. I need to forward my findings to Engineering.  
  
**END**