
Subject: Re: Problem using Split on carriage return
Posted by [mirek](#) on Tue, 10 Jul 2007 21:26:20 GMT
[View Forum Message](#) <> [Reply to Message](#)

Well, it is not U++, it is M\$. Back in MSDOS days, they have decided that line are to be ended with "\r\n" and this stupid idea is still alive.

Standard C library solves this problem by intruducing "text" and "binary" streams (text streams remove/insert \r on read/write), but IMO that just adds to confusion.

U++ general take on the problem is to ignore all \r characters on input and insert \r before each \n on output - but both is to be handled by the code.

In your case you have many options. One simple is to filter out all \r out before Split:

```
int NoCr(int c) { return c == '\r' ? 0 : c; }
```

```
Split(Filter(s, NoCr), '\n')
```

As you seem to ignore empty lines, another simple solution (with a bit better performance) is to treat \r like \n (because it will just result in empty line, which is ignored anyway):

```
int CrOrLf(int c) { return c == '\r' || c == '\n' ? c : 0; }
```

```
Split(s, CrOrLf);
```

Mirek
[/code]
