Subject: Re: Faster way to search/split a string?
Posted by Jeremy Bailin on Wed, 23 Nov 2011 16:02:23 GMT
View Forum Message <> Reply to Message

On 11/23/11 7:13 AM, Rob wrote:
> I was hoping that someone might have a cleverer way of approaching
> this problem.
>
> The following command is the bootleneck in my code:
>
> row_of_data=strsplit(all_rows[(where(stregex(all_rows, id, /boolean)
> eq 1))[0]],' ', /extract)
>
> I have a large text file with lots of columns of data (which I don't
> know exactly what the columns are until I've read them in). There are
> then say 10000 rows of this data.
>
> This is all read into one large string array (all_rows) which contains
> each row as a single very long string.
> The first 20 characters of the row contain a unique id which I need to
> search the rows for and then extract the entire matching row. This row
> then needs to be split up into it's columns (space delimited).
>
> Hopefully that all makes sense.
>
> The problem is having to do this 10000 times, (once for for each id)
> is very slow and the time to do all of the other stuff done in the
> code, reading, writing, some maths, etc is being dominated by this one
> line.
>
> Any thoughts or suggestions?
>
> Cheers
>
> Rob
>
> p.s. This needs to be GDL compatible as well which I think most
> solutions would be anyway.

This screams like something that should be outsourced to an external perl or python script. Dealing with strings is not IDL's forte, so why not use a better tool for that part if it's the bottleneck?

-Jeremy.