

---

Subject: GetValueAtLocation() method

Posted by [markjamie](#) on Tue, 11 Jun 2013 22:13:26 GMT

[View Forum Message](#) <> [Reply to Message](#)

---

Hi

I've been recently using the GetValueAtLocation() method in conjunction with the Surface function in IDL 8.2.2 but have been getting very mixed results.

When using p=surface(/test) together with p.getvalueatlocation(x, y), where x and Y are credible values it all works ok. When extending it's use to a real world example I encounter various errors.

My surface is a 2D array, essentially consisting of a number of time series, where the x dimension is usually positive (e.g. Time), the y dimension could be either positive or negative but is usually negative (e.g. Distance relative to a point), and the z dimension is a positive, scalar variable (e.g. Temperature).

I create a surface using p = surface(z, x, y) which gives good results. However when I try to use p.getvalueatlocation(10,-15, /interpolation), where 10 and -15 are valid x and Y coordinates on the surface, but perhaps not the individual time series (hence /interpolation), the method returns an error about subscripting (yloc) of a heap pointer variable is out of bounds. If the interpolation flag is removed another error occurs with similar results but a different variable name (nyloc or something v.similar).

Not sure if it makes any difference but the x dimension is always evenly spaced (e.g 1,2,3) but the y dimension may not be (e.g. 1,2,7,9).

Any thoughts on what is going on here? Can the GetValueAtLocation method deal with surface data where one dimension contains negative numbers?

Any help would be greatly appreciated as I'm baffled!

Mark

---