Subject: Re: Maps, overlaying, and !Pmulti (again)
Posted by David Fanning on Tue, 30 Apr 2013 13:26:41 GMT
View Forum Message <> Reply to Message

AMS writes:

> The situation is, I want to overlay two images (e.g. two separate satellite orbits) on one map projection, through sequential calls to map_set.
>
> If !p.multi=0, this seems to work fine:
>
> map_set
> (plot first orbit)
> map_set,/noerase
> (plot second orbit)

I'm not sure I understand what you are doing. You have two *different* map projections that you want to display in the same location in a graphics window? Why are you using two Map_Set commands?

Cheers,

David

--
David Fanning, Ph.D.
Fanning Software Consulting, Inc.
Coyote's Guide to IDL Programming: http://www.idlcoyote.com/
Sepore ma de ni thue. ("Perhaps thou speakest truth.")

Subject: Re: Maps, overlaying, and !Pmulti (again)
Posted by Andy Sayer on Tue, 30 Apr 2013 13:34:04 GMT
View Forum Message <> Reply to Message

Basically, I am writing a routine to map orbits of satellite data. I want to set it up so, if you want to overlay data from multiple satellites (with e.g. different spatial resolutions and swath positions) onto one image, you can call the routine twice and have the second orbit overlay the first. So in calling the plotting routine twice, map_set will inherently be called twice (although you'd pass the same map projection info).

Is there some better way to approach it? Hmmm. Perhaps I should instead write it so you're overplotting, it omits the second call to map_set?

Andrew

On Tuesday, April 30, 2013 9:26:41 AM UTC-4, David Fanning wrote:
> AMS writes:
>
>
>
>> The situation is, I want to overlay two images (e.g. two separate satellite orbits) on one map projection, through sequential calls to map_set.
>
>>
>
>> If !p.multi=0, this seems to work fine:
>
>>
>
>> map_set
>
>> (plot first orbit)
>
>> map_set,/noerase
>
>> (plot second orbit)
>
>
>
> I'm not sure I understand what you are doing. You have two *different*
>
> map projections that you want to display in the same location in a
>
> graphics window? Why are you using two Map_Set commands?
>
>
>
> Cheers,
>
>
>
> David
>
>
>
>
>
>
>
> --
>
> David Fanning, Ph.D.
>

> Fanning Software Consulting, Inc.
>
> Coyote's Guide to IDL Programming: http://www.idlcoyote.com/
>
> Sepore ma de ni thue. ("Perhaps thou speakest truth.")

Subject: Re: Maps, overlaying, and !Pmulti (again)
Posted by Andy Sayer on Tue, 30 Apr 2013 13:38:21 GMT
View Forum Message <> Reply to Message

Ok, think I realised the answer just after posting (as is often the way). David, your comment led me to it, thanks. I added a keyword to my routine which, if passed, bypasses the call to map_set. So you can overlay multiple images that way.

Andrew

On Tuesday, April 30, 2013 9:34:04 AM UTC-4, AMS wrote:
> Basically, I am writing a routine to map orbits of satellite data. I want to set it up so, if you want to overlay data from multiple satellites (with e.g. different spatial resolutions and swath positions) onto one image, you can call the routine twice and have the second orbit overlay the first. So in calling the plotting routine twice, map_set will inherently be called twice (although you'd pass the same map projection info).
>
>
>
> Is there some better way to approach it? Hmmm. Perhaps I should instead write it so you're overplotting, it omits the second call to map_set?
>
>
>
> Andrew
>
>
>
> On Tuesday, April 30, 2013 9:26:41 AM UTC-4, David Fanning wrote:
>
>> AMS writes:
>
>>
>
>>
>
>>
>
>>> The situation is, I want to overlay two images (e.g. two separate satellite orbits) on one map projection, through sequential calls to map_set.
>

>>
>
>>>
>
>>
>
>>> If !p.multi=0, this seems to work fine:
>
>>
>
>>>
>
>>
>
>>> map_set
>
>>
>
>>> (plot first orbit)
>
>>
>
>>> map_set,/noerase
>
>>
>
>>> (plot second orbit)
>
>>
>
>>
>
>>
>
>> I'm not sure I understand what you are doing. You have two *different*
>
>>
>
>> map projections that you want to display in the same location in a
>
>>
>
>> graphics window? Why are you using two Map_Set commands?
>
>>
>
>>
>

>>
>
>> Cheers,
>
>>
>
>>
>
>>
>
>> David
>
>>
>
>>
>
>>
>
>>
>
>>
>
>>
>
>>
>
>> --
>
>>
>
>> David Fanning, Ph.D.
>
>>
>
>> Fanning Software Consulting, Inc.
>
>>
>
>> Coyote's Guide to IDL Programming: http://www.idlcoyote.com/
>
>>
>
>> Sepore ma de ni thue. ("Perhaps thou speakest truth.")

Subject: Re: Maps, overlaying, and !Pmulti (again)
Posted by David Fanning on Tue, 30 Apr 2013 13:41:19 GMT

AMS writes:

> Basically, I am writing a routine to map orbits of satellite data. I want to set it up so, if you want to overlay data from multiple satellites (with e.g. different spatial resolutions and swath positions) onto one image, you can call the routine twice and have the second orbit overlay the first. So in calling the plotting routine twice, map_set will inherently be called twice (although you'd pass the same map projection info).
>
> Is there some better way to approach it? Hmmm. Perhaps I should instead write it so you're overplotting, it omits the second call to map_set?

Well, frankly, I think it is the "plotting" that is advancing !P.Multi, not the MAP_SET. So, yes, I think overplotting with a routine that doesn't "erase" the window would be a better idea. If you want to overplot "after the fact", then you can just save the normal system variables that get set by a plot operation (or, in your case, by MAP_SET) and restore them when you want to overplot.

Cheers,

David

--
David Fanning, Ph.D.
Fanning Software Consulting, Inc.
Coyote's Guide to IDL Programming: http://www.idlcoyote.com/
Sepore ma de ni thue. ("Perhaps thou speakest truth.")