Subject: Re: Version control for IDL software
Posted by Antonio Santiago on Tue, 22 Nov 2005 07:06:25 GMT
View Forum Message <> Reply to Message

Richard G. French wrote:
> I'm working with several students on the development of a set of data
> analysis tools in IDL, and I'd like to use some sort of version control for
> the IDL software. The programs reside on an OSX server but are accessible
> from a variety of other Mac machines. I'm running IDL under Mac OSX.
>
> I'd appreciate hearing from UNIX folks about the relative merits of CVS and
> RCS, or other approaches, to keeping track of versions of procedures and
> functions that are called from other IDL routines, or if this is even
> possible within the CVS/RCS paradigm.

Rememver SVN. It is the big/new brother of CVS.
>
> If these tools don't do the job, I'd like advice on other ways to make sure
> that I have a record of the exact versions of program files called by a
> given large IDL program. I'm less worried about two users editing the same
> file at the same time, and more concerned with coming up with a sensible
> scheme whereby we can 'freeze' a given version of a routine and feel fairly
> confident that we know which version is used at any given time.
>
Yes, they can edit the same file and "merge" it later. You can restore a
previous version, as old as it exists in the repository, instead to
rewrite some modifications. You can create new branches of development
and tag a state as a "first test version".

> What I've done in the past is to use a version number in the name of the
> procedure or function - for example:
>
> Pro complicated_procedure_v2,arg1,arg2
> End
>
> which would then be called from another program as
> complicated_procedure_V2, arg1,arg2
>
> However, I don't bump up the version number very often, and it can be a
> nuisance to change other code to call this new version of the program.
>
> Another approach might be to adopt a convention whereby every function or
> procedure that I write in the future has a VERSION keyword that would let me
> determine what version is being used during a given run of the code. This
> seems a bit clumsy but perhaps some of you have adopted this or a better
> scheme.
>
> What makes life a bit complicated is that lots of my routines call other

> utility routines from nifty libraries contributed by many of you folks.

>

--

Antonio Santiago Pï¿½rez
(email: santiago<<at>>grahi.upc.edu)
(www: http://www.grahi.upc.edu/santiago)
(www: http://asantiago.blogsite.org)

GRAHI - Grup de Recerca Aplicada en Hidrometeorologia
Universitat Politï¿½cnica de Catalunya
