

PSRCHIVE / DSPSR

“

Software: PSRCHIVE / DSPSR (intertwined)

Key Type(s): Int, String

Depends on: Sigproc, TEMPO, TEMPO2

Change at: Compilation

Repo: <https://sourceforge.net/projects/psrchive/>

<https://sourceforge.net/projects/dspsr/>

Quick links:

- PSRCHIVE
 - Util/tempo/itoa.C
<https://sourceforge.net/p/psrchive/code/ci/master/tree/Util/tempo/itoa.C>
 - Base/Extensions/Pulsar/Telescopes.h
<https://sourceforge.net/p/psrchive/code/ci/master/tree/Base/Extensions/Pulsar/Telescopes.h>
 - Base/Extensions/Telescopes.C
<https://sourceforge.net/p/psrchive/code/ci/master/tree/Base/Extensions/Telescopes.C>
- DSPSR
 - Kernel/Formats/sigproc/SigProcObservation.C
<https://sourceforge.net/p/dspsr/code/ci/master/tree/Kernel/Formats/sigproc/SigProcObservation.C>

PSRCHIVE and DSPSR are heavily intertwined, and need multiple changes at compilation time to provide initial support for a telescope, with further changes possible by modifying your TEMPO(2) configurations at runtime.

This focuses on the base software and Sigproc reader/writers, other output types may require further changes(e.g., FITS can be modified in the DSPSR `Kernel/Formats/FITS/...` path).

You will need to chose a site or telescope name, as well as be able to characterise your type of telescope.

- PSRCHIVE

- Util/tempo/itoa.C: `static int default_aliases()` needs you to tie your TEMPO1 code to your chosen site / telescope name.

```
/**** Previous aliases ***/  
add_alias (TEMPO_CODE, "MYSITENAME");  
/**** Further aliases ***/
```

- The itoa switch statement or the tempo2 logic needs to be able to identify your telescope. An example is not provided for this step.
- Base/Extensions/Pulsar/Telescopes.h: A new function prototype following the signature `void Pulsar::Telescopes::MYSITENAME(Telescope *t)` inside the namespace "`Telescopes`"

```
/**** Previous Prototypes ***/  
void Pulsar::Telescopes::MYSITENAME(Telescope *t)  
/**** More prototypes ***/
```

- Base/Extensions/Telescopes.C: A new function with signature following the prototype you just defined

```
/**** Other telescope definitions ***/  
  
void Pulsar::Telescopes::MYSITENAME(Telescope *t)  
{  
    t->set_name ("MYSITENAME");  
    /*// You will need to describe the properties of your telescope as well, check other entries in this file for further details.  
}  
/**** Further telescope definitions ***/
```

- DSPSR

- Kernel/Formats/sigproc/SigProcObservation.C: static `std::string get_sigproc_telescope_name (int _id)` needs your Sigproc site identifier and name.

```
/**** Previous Sigproc ID cases ***/  
case MY_SIGPROC_SITE_ID:
```

```
[return "MYSITENAME";  
/** Further Sigproc ID cases **/
```

- Kernel/Formats/sigproc/SigProcObservation.C: static int get_sigproc_telescope_id (string name) needs your Sigproc site name and TEMPO1/ITOA identifier.

```
[/** Other site code ifs **/  
[else if (itoa == "MY_TEMPO_CODE") return MY_SIGPROC_SITE_ID;  
[/** Further site code ifs **/
```

- Kernel/Formats/sigproc/SigProcObservation.C: static std::string get_sigproc_machine_name (int _id, int _telescope) needs your Sigproc processing node identifier and name.

```
[/** Previous Sigproc ID cases **/  
[case MY_SIGPROC_PROCESSING_NODE_ID:  
[return "MYNODENAME";  
/** Further Sigproc ID cases **/
```

- Kernel/Formats/sigproc/SigProcObservation.C: void dsp::SigProcObservation::unload_global () needs your Sigproc processing node identifier and name.

```
[/** Other processing node ID ifs **/  
[else if (get_machine().compare("MYNODENAME")==0) machine_id=MY_NODE_ID;  
[/** Further processing node ID ifs **/
```

Revision #2

Created 13 July 2021 13:54:56 by David

Updated 16 July 2021 16:45:45 by David