

NAME

unw_get_proc_info -- get info on current procedure

SYNOPSIS

```
#include <libunwind.h>
```

```
int unw_get_proc_info(unw_cursor_t *cp, unw_proc_info_t *pip);
```

DESCRIPTION

The `unw_get_proc_info()` routine returns auxiliary information about the procedure that created the stack frame identified by argument `cp`. The `pip` argument is a pointer to a structure of type `unw_proc_info_t` which is used to return the information. The `unw_proc_info_t` has the following members:

`unw_word_t start_ip`

The address of the first instruction of the procedure. If this address cannot be determined (e.g., due to lack of unwind information), the `start_ip` member is cleared to 0.

`unw_word_t end_ip`

The address of the first instruction *beyond* the end of the procedure. If this address cannot be determined (e.g., due to lack of unwind information), the `end_ip` member is cleared to 0.

`unw_word_t lsda`

The address of the language-specific data area (LSDA). This area normally contains language-specific information needed during exception handling. If the procedure has no such area, this member is cleared to 0.

`unw_word_t handler`

The address of the exception handler routine. This is sometimes called the *personality* routine. If the procedure does not define a personality routine, the `handler` member is cleared to 0.

`unw_word_t gp`

The global pointer of the procedure. On platforms that do not use a global pointer, this member may contain an undefined value. On all other platforms, it must be set either to the correct global pointer value of the procedure or to 0 if the proper global pointer cannot be obtained for some reason.

`unw_word_t flags`

A set of flags. There are currently no target-independent flags. For the IA-64 target, the flag `UNW_PI_FLAG_IA64_RBS_SWITCH` is set if the procedure may switch the register backing

store.

int format

The format of the unwind info for this procedure. If the unwind info consists of dynamic procedure info, format is equal to `UNW_INFO_FORMAT_DYNAMIC`. If the unwind info consists of a (target-specific) unwind table, it is equal to `UNW_INFO_FORMAT_TABLE`. All other values are reserved for future use by libunwind. This member exists for use by the `find_proc_info()` callback (see `unw_create_addr_space(3libunwind)`). The `unw_get_proc_info()` routine may return an undefined value in this member.

int unwind_info_size

The size of the unwind info in bytes. This member exists for use by the `find_proc_info()` callback (see `unw_create_addr_space(3libunwind)`). The `unw_get_proc_info()` routine may return an undefined value in this member.

void *unwind_info

The pointer to the unwind info. If no unwind info is available, this member must be set to `NULL`. This member exists for use by the `find_proc_info()` callback (see `unw_create_addr_space(3libunwind)`). The `unw_get_proc_info()` routine may return an undefined value in this member.

Note that for the purposes of libunwind, the code of a procedure is assumed to occupy a single, contiguous range of addresses. For this reason, it is always possible to describe the extent of a procedure with the `start_ip` and `end_ip` members. If a single function/routine is split into multiple, discontiguous pieces, libunwind will treat each piece as a separate procedure.

RETURN VALUE

On successful completion, `unw_get_proc_info()` returns 0. Otherwise the negative value of one of the error codes below is returned.

THREAD AND SIGNAL SAFETY

`unw_get_proc_info()` is thread safe. If cursor `cp` is in the local address space, this routine is also safe to use from a signal handler.

ERRORS

`UNW_EUNSPEC`

An unspecified error occurred.

`UNW_ENOINFO`

Libunwind was unable to locate unwind info for the procedure.

UNW_EBADVERSION

The unwind info for the procedure has version or format that is not understood by libunwind.

In addition, `unw_get_proc_info()` may return any error returned by the `access_mem()` callback (see `unw_create_addr_space(3libunwind)`).

SEE ALSO

`libunwind(3libunwind)`, `unw_create_addr_space(3libunwind)`, `unw_get_proc_name(3libunwind)`

AUTHOR

David Mosberger-Tang

Email: dmosberger@gmail.com

WWW: <http://www.nongnu.org/libunwind/>.