

**NAME**

unw\_reg\_states\_iterate -- get register state info on current procedure

**SYNOPSIS**

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

```
int unw_reg_states_iterate(unw_cursor_t *cp, unw_reg_states_callbackcb, void *token);
```

**DESCRIPTION**

The `unw_reg_states_iterate()` routine provides information about the procedure that created the stack frame identified by argument `cp`. The `cb` argument is a pointer to a function of type `unw_reg_states_callback` which is used to return the information. The function `unw_reg_states_callback` has the following definition:

```
int ( *unw_reg_states_callback)(void *token, void *reg_states_data, size_t reg_states_data_size,
unw_word_t start_ip, unw_word_t end_ip);
```

The callback function may be invoked several times for each call of `unw_reg_states_iterate`. Each call is associated with an instruction address range and a set of instructions on how to update register values when returning from the procedure in that address range. For each invocation, the arguments to the callback function are:

`void * token`

The token value passed to `unw_reg_states_callback`.

`void * reg_states_data`

A pointer to data about updating register values. This data, or a copy of it, can be passed to `unw_apply_reg_state`.

`int reg_states_data_size`

The size of the register update data.

`unw_word_t start_ip`

The address of the first instruction of the address range.

`unw_word_t end_ip`

The address of the first instruction *beyond* the end of the address range.

**RETURN VALUE**

On successful completion, `unw_reg_states_iterate()` returns 0. If the callback function returns a nonzero

value, that indicates failure and the function returns immediately. Otherwise the negative value of one of the error codes below is returned.

### **THREAD AND SIGNAL SAFETY**

`unw_reg_states_iterate()` 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_reg_states_iterate()` may return any error returned by the `access_mem()` call-back (see `unw_create_addr_space(3libunwind)`).

### **SEE ALSO**

`libunwind(3libunwind)`, `unw_apply_reg_state(3libunwind)`

### **AUTHOR**

David Mosberger-Tang

Email: [dmosberger@gmail.com](mailto:dmosberger@gmail.com)

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