

**NAME**

**scr\_dump**, **scr\_restore**, **scr\_init**, **scr\_set** - read (write) a **curses** screen from (to) a file

**SYNOPSIS**

```
#include <curses.h>
```

```
int scr_dump(const char *filename);
int scr_restore(const char *filename);
int scr_init(const char *filename);
int scr_set(const char *filename);
```

**DESCRIPTION**

The **scr\_dump** routine dumps the current contents of the *virtual screen* to the file *filename*.

The **scr\_restore** routine sets the *virtual screen* to the contents of *filename*, which must have been written using **scr\_dump**. The next call to **doupdate** restores the *physical screen* to the way it looked in the dump file.

The **scr\_init** routine reads in the contents of *filename* and uses them to initialize the **curses** data structures about what the terminal currently has on its screen. If the data is determined to be valid, **curses** bases its next update of the screen on this information rather than clearing the screen and starting from scratch. **scr\_init** is used after **initscr** or a **system** call to share the screen with another process which has done a **scr\_dump** after its **endwin(3X)** call. The data is declared invalid

- ⊕ if the terminfo capabilities **rmcup** and **nrrmc** exist, also
- ⊕ if the terminal has been written to since the preceding **scr\_dump** call.

The **scr\_set** routine is a combination of **scr\_restore** and **scr\_init**. It tells the program that the information in *filename* is what is currently on the screen, and also what the program wants on the screen. This can be thought of as a screen inheritance function.

To read (write) a window from (to) a file, use the **getwin** and **putwin** routines [see **curs\_util(3X)**].

**RETURN VALUE**

All routines return the integer **ERR** upon failure and **OK** upon success.

X/Open defines no error conditions. In this implementation, each will return an error if the file cannot be opened.

`curs_scr_dump(3X)`

`curs_scr_dump(3X)`

## **NOTES**

Note that `scr_init`, `scr_set`, and `scr_restore` may be macros.

## **PORTABILITY**

The XSI Curses standard, Issue 4, describes these functions (adding the `const` qualifiers).

The SVr4 docs merely say under `scr_init` that the dump data is also considered invalid "if the time-stamp of the tty is old" but do not define "old".

## **SEE ALSO**

`curses(3X)`, `curs_initscr(3X)`, `curs_refresh(3X)`, `curs_util(3X)`, `scr_dump(5)`, `system(3)`

`curs_scr_dump(3X)`