

export_strip.py Documentation

Written 6/15/20 by Derek C. Richardson (dcr@umd.edu)

Last Update 6/25/20 by Derek C. Richardson (dcr@umd.edu)

Contents

1	OVERVIEW	1
2	OPTIONS	1
3	KNOWN ISSUES	1

1 OVERVIEW

The `export_strip.py` utility simplifies preparation of the `pkdgrav` codebase for export by filtering out macro-defined code blocks from the source. For example, running the script with the argument `DEM` strips all elements of the soft-sphere implementation. The utility creates a directory (named `export` by default) containing the stripped code ready for packaging into a tar ball or zip file. Note the code must be run at the top of the codebase, which can be the local `git` working directory or either a clone of the remote repository or a published release (version 2.5.0 or later).

Any valid macro(s) may be provided in the argument list (use `export_strip.py --help` for a full list of options). Normally however these would be macros specified in `Makefile.in` as compilation options, e.g., `WALLS`, `AGGS`, `DEM`, etc.

Certain macros are automatically stripped. These include `NUMREC` (the *Numerical Recipes* routines, which cannot be distributed) along with older or more experimental macros that are not meant to be included in any distribution.

When the utility has completed its operation, a message with further instructions is displayed, as some final steps must be completed by hand, such as editing `Makefile.in` to remove the stripped macros.

It is intended that `export_strip.py` be used only by code developers. The script and this documentation are *not* included in the export.

2 OPTIONS

<code>-h</code> or <code>--help</code>	Show usage message.
<code>-e EXPORT</code> or <code>--export EXPORT</code>	Use <code>EXPORT</code> as export directory (default <code>export</code>).
<code>-v</code> or <code>--verbose</code>	Display more verbose output.

3 KNOWN ISSUES

1. The `export_strip.py` utility was introduced at Release 2.5.0 and will not work completely for prior releases (since certain code modifications were required to make the script work well).
2. Any new code macros added subsequently may require an update to `export_strip.py` in each instance if it is desired to strip out the new code for export.
3. Sometimes, particularly when more than one macro test occurs at once (e.g., the conditional “`if defined(WALLS) || defined(AGGS)`”), references to a macro may remain, but all code specific *only* to that macro will still be stripped.
4. All soft links get converted to regular files on export, causing some minor duplication.
5. The script can only do so much. For proper public distribution, items like README files will need to be modified or added by hand, along with other specific instructions that are not normally part of the codebase.