1 Apache::SubRequest - Perl API for Apache subrequests

1.1 Synopsis

```
  use Apache::SubRequest ();
```

META: to be completed

1.2 Description

META: to be completed

1.3 API

`Apache::SubRequest` provides the following functions and/or methods:

1.3.1 ***`DESTROY`***

META: Autogenerated - needs to be reviewed/completed

Free the memory associated with a sub request

```
  $r->DESTROY();
```

- **arg1: `$r` (`Apache::RequestRec`)**

  The sub request to finish

- **ret: no return value**

1.3.2 ***`internal_fast_redirect`***

META: Autogenerated - needs to be reviewed/completed

Redirect the current request to a sub_req, merging the pools

```
  $r->internal_fast_redirect($sub_req);
```

- **arg1: `$r` (`Apache::RequestRec`)**

  The current request

- **arg2: `$sub_req` (string)**

  A subrequest created from this request

- **ret: no return value**

1.3.3 *internal_redirect*

META: Autogenerated - needs to be reviewed/completed

Then there’s the case that you want some other request to be served as the top-level request INSTEAD of what the client requested directly. If so, call this from a handler, and then immediately return OK.

Redirect the current request to some other uri

```
  $r->internal_redirect($new_uri);
```

- **arg1: `$r` (`Apache::RequestRec`)**

  The current request

- **arg2: `$new_uri` (string)**

  The URI to replace the current request with

- **ret: no return value**

1.3.4 *internal_redirect_handler*

META: Autogenerated - needs to be reviewed/completed

This function is designed for things like actions or CGI scripts, when using AddHandler, and you want to preserve the content type across an internal redirect.

```
  $r->internal_redirect_handler($new_uri);
```

- **arg1: `$r` (`Apache::RequestRec`)**

  The current request

- **arg2: `$new_uri` (string)**

  The URI to replace the current request with.

- **ret: no return value**

1.3.5 *lookup_dirent*

META: Autogenerated - needs to be reviewed/completed

Create a sub request for the given apr_dir_read result. This sub request can be inspected to find information about the requested file

```
$lr = $r->lookup_dirent($finfo, $subtype, $next_filter);
```

- **arg1: `$r` (`Apache::RequestRec`)**

  The current request

- **arg2: `$finfo` (`APR::Finfo`)**

  The apr_dir_read result to lookup

- **arg3: `$subtype` (integer)**

  What type of subrequest to perform, one of;

  ```
  Apache::SUBREQ_NO_ARGS     ignore r->args and r->path_info
  Apache::SUBREQ_MERGE_ARGS  merge  r->args and r->path_info
  ```

- **arg4: `$next_filter` (integer)**

  The first filter the sub_request should use. If this is NULL, it defaults to the first filter for the main request

- **ret: `$lr` (`Apache::RequestRec`)**

  The new request record

1.3.6 `lookup_file`

META: Autogenerated - needs to be reviewed/completed

Create a sub request for the given file. This sub request can be inspected to find information about the requested file

```
$ret = $r->lookup_file($new_file, $next_filter);
```

- **arg1: `$r` (`Apache::RequestRec`)**

  The current request

- **arg2: `$new_file` (string)**

  The URI to lookup

- **arg3: `$next_filter` (`Apache::Filter`)**

  The first filter the sub_request should use. If this is NULL, it defaults to the first filter for the main request

- **ret: `$ret` (`Apache::SubRequest`)**

The new request record

1.3.7 *`lookup_uri`*

META: Autogenerated - needs to be reviewed/completed

Create a sub request from the given URI. This sub request can be inspected to find information about the requested URI.

```
$ret = $r->lookup_uri($new_file, $next_filter);
```

- **arg1: `$r` (`Apache::RequestRec`)**

  The current request

- **arg2: `$new_file` (string)**

  The URI to lookup

- **arg3: `$next_filter` (`Apache::Filter`)**

  The first filter the sub_request should use. If this is NULL, it defaults to the first filter for the main request

- **ret: `$ret` (`Apache::SubRequest`)**

  The new request record

1.3.8 *`lookup_method_uri`*

META: Autogenerated - needs to be reviewed/completed

Create a sub request for the given URI using a specific method. This sub request can be inspected to find information about the requested URI

```
$ret = $r->lookup_method_uri($method, $new_file, $next_filter);
```

- **arg1: `$r` (`Apache::RequestRec`)**

  The current request

- **arg2: `$method` (string)**

  The method to use in the new sub request

- **arg3: `$new_file` (string)**

  The URI to lookup

- **arg4: `$next_filter` (`Apache::Filter`)**

  The first filter the sub_request should use. If this is NULL, it defaults to the first filter for the main request

- **ret: `$ret` (`Apache::SubRequest`)**

  The new request record

1.4 See Also

mod_perl 2.0 documentation.

1.5 Copyright

mod_perl 2.0 and its core modules are copyrighted under The Apache Software License, Version 1.1.

1.6 Authors

The mod_perl development team and numerous contributors.

Table of Contents: