

1 Apache::Connection - Perl API for Apache connection object

1.1 Synopsis

```
use Apache::Connection ();
```

META: to be completed

1.2 Description

META: to be completed

1.3 API

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

1.3.1 aborted

Check whether the connection is still open

```
$status = $c->aborted();
```

- **arg1: \$c (Apache::Connection)**
- **ret: \$status (number)**

true if the connection has been aborted, false if still open

1.3.2 base_server

Physical vhost this connection came in on

```
$base_server = $c->base_server();
```

- **arg1: \$c (Apache::Connection)**
- **ret: \$base_server (Apache::Server)**

1.3.3 bucket_alloc

META: Autogenerated - needs to be reviewed/completed

The bucket allocator to use for all bucket/brigade creations

```
$ba = $c->bucket_alloc();
```

- **arg1: \$c (Apache::Connection)**
- **ret: \$ba (APR::BucketAlloc)**

1.3.4 conn_config

META: Autogenerated - needs to be reviewed/completed

Notes on **this** connection

```
$ret = $c->conn_config();
```

- **arg1: \$c (Apache::Connection)**
- **ret: \$ret (Apache::ConfVector)**

1.3.5 id

ID of this connection; unique at any point in time

```
$id = $c->id();
```

- **arg1: \$c (Apache::Connection)**
- **ret: \$id (integer)**

1.3.6 input_filters

A list of input filters to be used for this connection

```
$input_filters = $c->input_filters();
```

- **arg1: \$c (Apache::Connection)**
- **ret: \$input_filters (Apache::Filter)**

The first filter in the connection input filters chain.

1.3.7 keepalive

Are we going to keep the connection alive for another request?

```
$status = $c->keepalive();
```

- **arg1: \$c (Apache::Connection)**
- **ret: \$status (integer)**

1.3.8 local_addr

Get this connection's local socket address

```
$sa = $c->local_addr();
```

- **arg1:** `$c` (**Apache::Connection**)
- **ret:** `$sa` (**APR::SockAddr**)

1.3.9 local_host

used for `ap_get_server_name` when `UseCanonicalName` is set to `DNS` (ignores setting of `Host-nameLookups`)

```
$local_host = $c->local_host();
```

- **arg1:** `$c` (**Apache::Connection**)
- **ret:** `$local_host` (**string**)

1.3.10 local_ip

server IP address

```
$local_ip = $c->local_ip();
```

- **arg1:** `$c` (**Apache::Connection**)
- **ret:** `$local_ip` (**string**)

1.3.11 notes

META: Autogenerated - needs to be reviewed/completed

send note from one module to another, must remain valid for all requests on this conn

```
$c->notes($notes);  
$notes = $c->notes();
```

- **arg1:** `$c` (**Apache::Connection**)
- **arg2:** `$notes` (**APR::Table**)
- **ret:**

1.3.12 output_filters

META: Autogenerated - needs to be reviewed/completed

A list of output filters to be used for this connection

```
$output_filters = $c->output_filters();
```

- **arg1:** `$c` (**Apache::Connection**)
- **ret:** `$output_filters` (**Apache::Filter**)

The first filter in the connection output filters chain.

1.3.13 pool

Pool associated with this connection

```
$p = $c->pool();
```

- **arg1:** `$c` (**Apache::Connection**)
- **ret:** `$p` (**APR::Pool**)

1.3.14 remote_addr

Get this connection's remote socket address

```
$sa = $c->remote_addr();
```

- **arg1:** `$c` (**Apache::Connection**)
- **ret:** `$sa` (**APR::SockAddr**)

1.3.15 remote_ip

Client's IP address

```
$remote_ip = $c->remote_ip();
```

- **arg1:** `$c` (**Apache::Connection**)
- **ret:** `$remote_ip` (**string**)

1.3.16 remote_host

Client's DNS name, if known. NULL if DNS hasn't been checked, "" if it has and no address was found. N.B. Only access this through `get_remote_host()`

```
$remote_host = $c->remote_host();
```

- **arg1:** `$c` (**Apache::Connection**)
- **ret:** `$remote_host` (**string**)

1.3.17 remote_logname

Only ever set if doing rfc1413 lookups. N.B. Only access this through `get_remote_logname()`

```
$remote_logname = $c->remote_logname();
```

1.4 See Also

- **arg1:** `$c` (**Apache::Connection**)
- **ret:** `$remote_logname` (**string**)

1.3.18 sbh

META: Autogenerated - needs to be reviewed/completed

handle to scoreboard information for this connection

```
$sbh = $c->sbh();
```

- **arg1:** `$c` (**Apache::Connection**)
- **ret:** `$sbh` (**XXX**)

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:

1	Apache::Connection - Perl API for Apache connection object	1
1.1	Synopsis	2
1.2	Description	2
1.3	API	2
1.3.1	aborted	2
1.3.2	base_server	2
1.3.3	bucket_alloc	2
1.3.4	conn_config	3
1.3.5	id	3
1.3.6	input_filters	3
1.3.7	keepalive	3
1.3.8	local_addr	3
1.3.9	local_host	4
1.3.10	local_ip	4
1.3.11	notes	4
1.3.12	output_filters	4
1.3.13	pool	5
1.3.14	remote_addr	5
1.3.15	remote_ip	5
1.3.16	remote_host	5
1.3.17	remote_logname	5
1.3.18	sbh	6
1.4	See Also	6
1.5	Copyright	6
1.6	Authors	6