

NAME

Mojo::Server::FastCGI – FastCGI Server

SYNOPSIS

```
use Mojo::Server::FastCGI;

my $fcgi = Mojo::Server::FastCGI->new;
$fcgi->on_request(sub {
    my ($self, $tx) = @_;

    # Request
    my $method = $tx->req->method;
    my $path   = $tx->req->url->path;

    # Response
    $tx->res->code(200);
    $tx->res->headers->content_type('text/plain');
    $tx->res->body("{$method} request for $path!");

    # Resume transaction
    $tx->resume;
});

$fcgi->run;
```

DESCRIPTION

Mojo::Server::FastCGI is a portable pure-Perl FastCGI implementation as described in the [FastCGI Specification](#).

See Mojolicious::Guides::Cookbook for deployment recipes.

ATTRIBUTES

Mojo::Server::FastCGI inherits all attributes from Mojo::Server.

METHODS

Mojo::Server::FastCGI inherits all methods from Mojo::Server and implements the following new ones.

```
accept_connection
    my $c = $fcgi->accept_connection;
    Accept FastCGI connection.

read_record
    my ($type, $id, $body) = $fcgi->read_record($c);
    Parse FastCGI record.

read_request
    my $tx = $fcgi->read_request($c);
    Parse FastCGI request.

role_name
    my $name = $fcgi->role_name(3);
    FastCGI role name.

role_number
    my $number = $fcgi->role_number('FILTER');
    FastCGI role number.

run
    $fcgi->run;
    Start FastCGI.
```



```
type_name
    my $name = $fcgi->type_name(5);
    FastCGI type name.

type_number
    my $number = $fcgi->type_number('STDIN');
    FastCGI type number.

write_records
    $fcgi->write_record($c, 'STDOUT', $id, 'HTTP/1.1 200 OK');
    Write FastCGI record.

write_response
    $fcgi->write_response($tx);
    Write FastCGI response.
```

DEBUGGING

You can set the MOJO_FASTCGI_DEBUG environment variable to get some advanced diagnostics information sent to the Mojo logger as debug messages.

MOJO_FASTCGI_DEBUG=1

SEE ALSO

Mojolicious, Mojolicious::Guides, <<http://mojolicio.us>>.

