

**NAME**

TAP::Formatter::Base - Base class for harness output delegates

**VERSION**

Version 3.36

**DESCRIPTION**

This provides console orientated output formatting for TAP::Harness.

**SYNOPSIS**

```
use TAP::Formatter::Console;
my $harness = TAP::Formatter::Console->new( \%args );
```

**METHODS****Class Methods**

*new*

```
my %args = (
    verbose => 1,
)
my $harness = TAP::Formatter::Console->new( \%args );
```

The constructor returns a new `TAP::Formatter::Console` object. If a `TAP::Harness` is created with no formatter a `TAP::Formatter::Console` is automatically created. If any of the following options were given to `TAP::Harness->new` they will be passed to this constructor which accepts an optional hashref whose allowed keys are:

- `verbosity`  
Set the verbosity level.
- `verbose`  
Printing individual test results to STDOUT.
- `timer`  
Append run time for each test to output. Uses `Time::HiRes` if available.
- `failures`  
Show test failures (this is a no-op if `verbose` is selected).
- `comments`  
Show test comments (this is a no-op if `verbose` is selected).
- `quiet`  
Suppressing some test output (mostly failures while tests are running).
- `really_quiet`  
Suppressing everything but the tests summary.
- `silent`  
Suppressing all output.
- `errors`  
If parse errors are found in the TAP output, a note of this will be made in the summary report. To see all of the parse errors, set this argument to true:  

```
errors => 1
```
- `directives`  
If set to a true value, only test results with directives will be displayed. This overrides other settings such as `verbose`, `failures`, or `comments`.

- `stdout`  
A filehandle for catching standard output.
- `color`  
If defined specifies whether color output is desired. If `color` is not defined it will default to color output if color support is available on the current platform and output is not being redirected.
- `jobs`  
The number of concurrent jobs this formatter will handle.
- `show_count`  
Boolean value. If false, disables the X/Y test count which shows up while tests are running.

Any keys for which the value is `undef` will be ignored.

#### *prepare*

Called by [Test::Harness](#) before any test output is generated.

This is an advisory and may not be called in the case where tests are being supplied to [Test::Harness](#) by an iterator.

#### *open\_test*

Called to create a new test session. A test session looks like this:

```
my $session = $formatter->open_test( $test, $parser );
while ( defined( my $result = $parser->next ) ) {
    $session->result($result);
    exit 1 if $result->is_bailout;
}
$session->close_test;
```

#### *summary*

```
$harness->summary( $aggregate );
```

`summary` prints the summary report after all tests are run. The first argument is an aggregate to summarise. An optional second argument may be set to a true value to indicate that the summary is being output as a result of an interrupted test run.