

**NAME**

"Test::Future::Deferred" – a future which completes later

```
my $future = Test::Future::Deferred->done_later( 1, 2, 3 );

# Future is not ready yet

my @result = $future->get;
```

**DESCRIPTION**

This subclass of Future provides two new methods and an implementation of the `await` interface, which allows the futures to appear pending at first, but then to complete when `get` is called at the toplevel on one of them.

This behaviour is useful in unit tests to check that behaviour of a module under test is correct even with non-immediate futures, as it allows a future to easily be constructed that will complete “soon”, but not yet, without needing an event loop.

Because these futures provide their own `await` method, they shouldn't be mixed in the same program with other kinds of futures from real event systems or similar.

**METHODS****done\_later**

```
$f->done_later( @args )
```

Equivalent to invoking the regular `done` method as part of the `await` operation called on the toplevel future. This makes the future complete with the given result, but only when `get` is called.

**fail\_later**

```
$f->fail_later( $message, $category, @details )
```

Equivalent to invoking the regular `fail` method as part of the `await` operation called on the toplevel future. This makes the future complete with the given failure, but only when `get` is called. As the `failure` method also waits for completion of the future, then it will return the failure message given here also.

**AUTHOR**

Paul Evans <leonerd@leonerd.org.uk>