

```
detail::simple_state
 _base_type< X2 , Transition
EventBaseTest ::inner_context
 _type, mpl::list<> >::type
```

```
detail::simple_state
 _base_type< X2 , DeferralEvent
BaseTest ::inner_context_type,
 mpl::list<> >::type
```

boost::statechart::
simple_state< X2, Transition
EventBaseTest >

- pContext_

+ outermost_context()
+ outermost_context()
+ context()
+ context()
+ state_cast()
+ state_downcast()
+ state_begin()
+ state_end()
+ post_event()
+ post_event()
+ discard_event()
+ forward_event()
+ defer_event()
+ transit()
+ transit()
+ terminate()
+ clear_shallow_history()
+ clear_deep_history()
+ triggering_event()
+ BOOST_STATIC_ASSERT()
+ BOOST_STATIC_ASSERT()
+ operator new()
+ operator delete()
+ outermost_context_base()
+ outermost_context_base()
+ outer_state_ptr()
+ react_Impl()
+ exit_Impl()
+ exit_Impl()
+ set_outermost_unstable
_state()
+ context_ptr()
+ set_context()
+ store_deep_history_Impl()
+ initial_deep_construct()
+ deep_construct()
+ shallow_construct()
+ deep_construct_inner()
simple_state()
~simple_state()
- transit_Impl()
- local_react()
- check_store_shallow
_history()
- check_store_deep_history()

boost::statechart::
simple_state< X2, Deferral
EventBaseTest >

- pContext_

+ outermost_context()
+ outermost_context()
+ context()
+ context()
+ state_cast()
+ state_downcast()
+ state_begin()
+ state_end()
+ post_event()
+ post_event()
+ discard_event()
+ forward_event()
+ defer_event()
+ transit()
+ transit()
+ terminate()
+ clear_shallow_history()
+ clear_deep_history()
+ triggering_event()
+ BOOST_STATIC_ASSERT()
+ BOOST_STATIC_ASSERT()
+ operator new()
+ operator delete()
+ outermost_context_base()
+ outermost_context_base()
+ outer_state_ptr()
+ react_Impl()
+ exit_Impl()
+ exit_Impl()
+ set_outermost_unstable
_state()
+ context_ptr()
+ set_context()
+ store_deep_history_Impl()
+ initial_deep_construct()
+ deep_construct()
+ shallow_construct()
+ deep_construct_inner()
simple_state()
~simple_state()
- transit_Impl()
- local_react()
- check_store_shallow
_history()
- check_store_deep_history()

