

Doc No: WG21 N3776
Date: 2013-09-23
Reply to: Herb Sutter (hsutter@microsoft.com)
Subgroup: SG1 – Concurrency
Previous Version: N3773

Wording for ~future

Herb Sutter

This paper provides proposed wording to implement a positive SG1 straw poll to clarify that ~future and ~shared_future don't block except possibly in the presence of async.

Change 30.6.4/5 as follows:

- 5 When an asynchronous return object or an asynchronous provider is said to release its shared state, it means:
 - if the return object or provider holds the last reference to its shared state, the shared state is destroyed; ~~and~~
 - the return object or provider gives up its reference to its shared state; and
 - these actions will not block for the shared state to become ready, except that it may block if all of the following are true: the shared state was created by a call to std::async, the shared state is not yet ready, and this was the last reference to the shared state.

In 30.6.8/4, add: [Note: If a future obtained from std::async is moved outside the local scope, other code that uses the future must be aware that the future's destructor may block for the shared state to become ready.—end note]