

try_lock Summary

- Scenario:
 - Data is big: `std::atomic<>::is_always_lock_free == false`
 - Failure to acquire the resource is ok
- Trade-off:
 - Non-real-time thread waits on real-time thread for access to the resource
 - Real-time thread will have to fail gracefully
- Examples:
 - Passing large data to the real-time thread for exclusive use
 - Audio samples, wavetables, filter coefficients etc.

Seriously, can you use a lock?

