

# Locking in ARC

Ulf Tigerstedt

# Mutexes

- Mostly `Glib::Mutex::Lock`

## Mutexes

- Mostly `Glib::Mutex::Lock`
- Some `Glib::Mutex lock()`

## Mutexes

- Mostly `Glib::Mutex::Lock`
- Some `Glib::Mutex lock()`
- Very few `Glib::RecMutex::Lock`

## Mutexes

- Mostly `Glib::Mutex::Lock`
- Some `Glib::Mutex lock()`
- Very few `Glib::RecMutex::Lock`
- Globus code uses globus locking

# Naming

- lock

# Naming

- lock
- lock\_

## Naming

- lock
- lock\_
- mutex, synchMutex



## Depreciation recursion

- `Glib::Mutex::Lock` is deprecated.

## Depreciation recursion

- `Glib::Mutex::Lock` is deprecated.
- Use `Glib::Threads::Mutex::Lock` instead.

## Depreciation recursion

- `Glib::Threads::Mutex::Lock` is deprecated.

## Depreciation recursion

- `Glib::Threads::Mutex::Lock` is deprecated.
- Please use `std::lock_guard` or `std::unique_lock` instead.

## Problems

- Clusters such as triolith@NSC has seen >180 seconds of lock wait, possibly Xen a part of this

## Problems

- Clusters such as triolith@NSC has seen >180 seconds of lock wait, possibly Xen a part of this
- ARC 5.3.1 is better than the previous versions (5.3.0 was completely broken).