

# NXLock

<b>Inherits From:</b>	Object
<b>Conforms To:</b>	NXLock
<b>Declared In:</b>	machkit/NXLock.h

## Class Description

An NXLock is used to protect regions of code that can consume long periods of time, such as disk I/O or heavy computations. A lock is created once and is subsequently used to protect one or more regions of code. If a region of code is in use, an NXLock waits using the **condition\_wait()** function, so the thread doesn't busy-wait. The following example shows the use of an NXLock:

```
NXLock *theLock = [[NXLock alloc] init];           // done once!  
/* ... other code */  
[theLock lock];  
/* ... possibly a long time of fussing with global data... */  
[theLock unlock];
```

The NXConditionLock, NXLock, NXRecursiveLock, and NXSpinLock classes all implement the NXLock protocol with various features and performance characteristics; see the other class descriptions for more information.

## Instance Variables

None declared in this class.

## Method Types

Acquire or release a lock      – lock  
   – unlock

## Instance Methods

### **lock**

– **lock**

Waits until the lock isn't in use, using **condition\_wait()** if necessary, then grabs the lock.

### **unlock**

– **unlock**

Releases the lock, using **condition\_signal()** to signal the next party that the lock is available.