

**15-410**

***“What could possibly go wrong?”***

**Paradise Lost**  
**Sep. 21, 2007**

**Dave Eckhardt**

**Roger Dannenberg**

# Synchronization

## Cluster seats

- Who found this an issue?
  - How well do the Macs work?

# Synchronization

## Cluster seats

- Who found this an issue?
  - How well do the Macs work?

## A new space opening up

- West Wing dorm, ground floor
  - Open to anybody with a CMU i.d. card
- Mixture of machine types, work-area types
  - Experimental
- Planned opening Monday

# Outline

**When to use `if()` vs. `while()`**

# Consider the lowly worker thread

```
/* note: not a thrgrp_*() worker thread */  
void  
worker(void *ignored)  
{  
    workitem *work;  
    while (work = find_work())  
        perform(work);  
    thr_exit((void *) 0);  
}
```

# What's Wrong With This Picture?

```
workitem *  
find_work(void)  
{  
    workitem *w;  
    mutex_lock(&m);  
    if (going_out_of_business)  
        w = (workitem *) 0;  
    else  
        w = (workitem *) dequeue(q);  
    mutex_unlock(&m);  
    return (w);  
}
```

# Better?

```
mutex_lock(&m);  
if (going_out_of_business)  
    w = (workitem *) 0;  
else {  
    if (!(w = (workitem *) dequeue(q))) {  
        cond_wait(&new_work, &m);  
        w = (workitem *) dequeue(queue);  
    }  
}  
mutex_unlock(&m);  
return (w);
```

# What We Hope For

<i>find_work()</i>	<i>queue_work()</i>
<code>mutex_lock(&amp;m);</code>	
<code>if (!..dequeue(..))</code>	
<code>cond_wait(&amp;new, &amp;m);</code>	
	<code>mutex_lock(&amp;m);</code>
	<code>enqueue(...)</code>
	<code>cond_signal(&amp;new);</code>
	<code>mutex_unlock(&amp;m);</code>
<code>w = dequeue(..);</code>	
<code>mutex_unlock(&amp;m);</code>	



# What Went Wrong?

**What went wrong?**

# What Went Wrong?

## What went wrong?

- Nothing!

# What Went Wrong?

**What went wrong?**

- **Nothing!**

**But what if there is *another* thread?**

# Not Exactly What We Hope For

<i>find_work()</i>	<i>queue_work()</i>	<i>find_work()</i>
<code>lock (&amp;m) ;</code>		
<code>if (!...deq(..))</code>		
<code>cwait(&amp;new, &amp;m) ;</code>		
	<code>lock (&amp;m) ;</code>	
	<code>enqueue (...)</code>	
	<code>csignal (&amp;new) ;</code>	
	<code>unlock (&amp;m) ;</code>	
		<code>lock (&amp;m) ;</code>
		<code>if (!...deq(..))</code>
		<code>unlock (&amp;m) ;</code>
<code>w = deq(..)...</code>		<code>return(w) ;</code>
<code>return (0) ;</code>		

# Have We Seen This Before?

## What went wrong?

- Protected world state wasn't ready for us
- We went to sleep
- Somebody prepared the world for us to run
- We ran
  - We *assumed* nobody else had run
  - We *assumed* the world state was still ready for us

# To “if()” Or Not To “if()”?

```
mutex_lock(&m);  
if (going_out_of_business)  
    w = (workitem *) 0;  
else {  
    while (!(w = (workitem *) dequeue(q)))  
        cond_wait(&new_work, &m);  
}  
mutex_unlock(&m);  
return (w);  
/* XXX still wrong! - rewrite after class */
```

# Summary

## **if() vs. while()**

- If somebody can revoke your happiness, you'd better check