

cse

## Scheduler Activations

THE UNIVERSITY OF NEW SOUTH WALES

Slides modified from Raymond Namyst, U. Bordeaux

cse

## Learning Outcomes

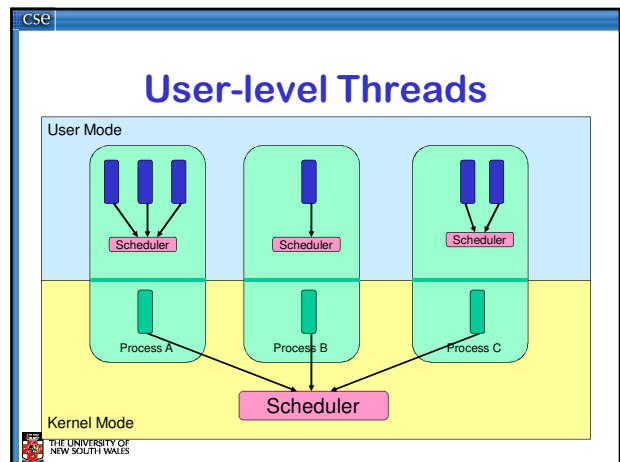
- An understanding of hybrid approaches to thread implementation
- A high-level understanding of scheduler activations, and how they overcome the limitations of user-level and kernel-level threads.

THE UNIVERSITY OF NEW SOUTH WALES

cse

- Thomas Anderson, Brian Bershad, Edward Lazowska, and Henry Levy. Scheduler Activations: Effective Kernel Support for the User-Level management of Parallelism. ACM Trans. on Computer Systems 10(1), February 1992, pp. 53-79.

THE UNIVERSITY OF NEW SOUTH WALES

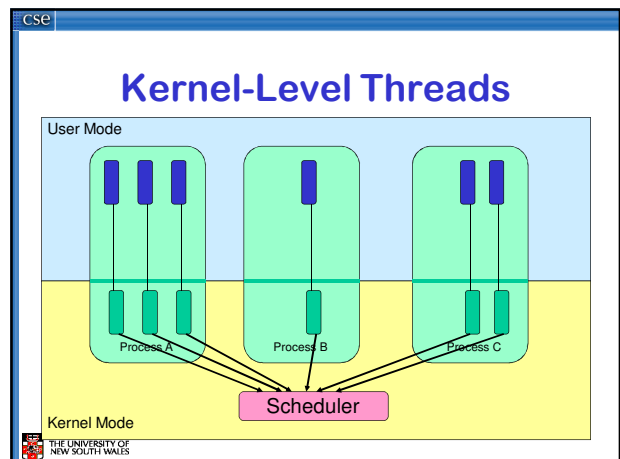


cse

## User-level Threads


- ✓ Fast thread management (creation, deletion, switching, synchronisation...)
- ✗ Blocking blocks all threads in a process
  - Syscalls
  - Page faults
- ✗ No thread-level parallelism on multiprocessor

THE UNIVERSITY OF NEW SOUTH WALES



## Kernel-level Threads


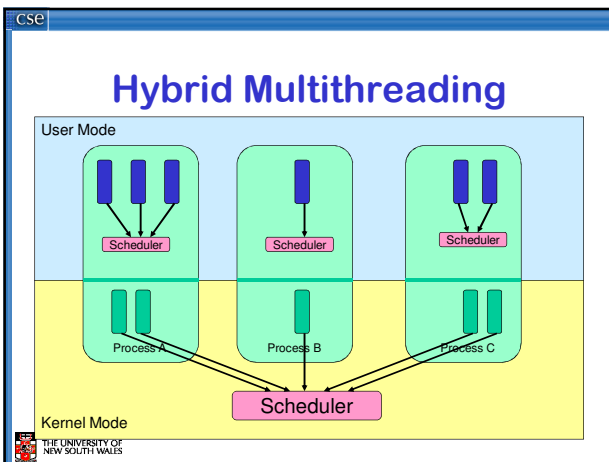
- ✗ Slow thread management (creation, deletion, switching, synchronisation...)
  - System calls
- ✓ Blocking blocks only the appropriate thread in a process
- ✓ Thread-level parallelism on multiprocessor



## Performance


Table I: Thread Operation Latencies (µsec.)

Operation	FastThreads	Topaz threads	Ultrix processes
Null Fork	34	948	11300
Signal-Wait	37	441	1840

## Hybrid Multithreading


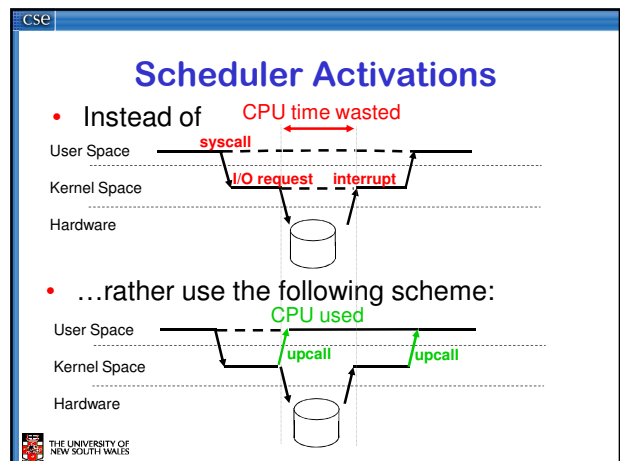
- ✓ Can get real thread parallelism on multiprocessor
- ✗ Blocking still a problem!!!



## Scheduler Activations


- First proposed by [Anderson et al. 91] K
- Idea: Both schedulers co-operate
  - User scheduler uses system calls
  - Kernel scheduler uses upcalls!
- Two important concepts
  - Upcalls
    - Notify the user-level of kernel scheduling events
  - Activations
    - A new structure to support upcalls and execution
      - approximately a kernel thread
    - As many running activations as (allocated) processors
    - Kernel controls activation creation and destruction

K  
V  
DDD

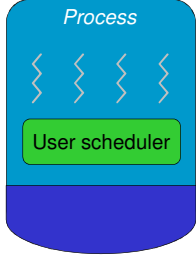

**Upcalls to User-level scheduler**

- **New**
  - Allocated a new virtual CPU
  - Can schedule a user-level thread
- **Preempted**
  - Deallocated a virtual CPU
  - Can schedule one less thread
- **Blocked**
  - Notifies thread has blocked
  - Can schedule another user-level thread
- **Unblocked**
  - Notifies a thread has become runnable
  - Must decided to continue current or unblocked thread



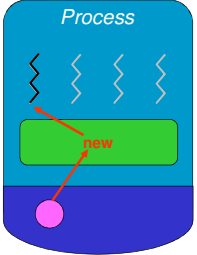

**Working principle**

- Blocking syscall scenario on 2 processors

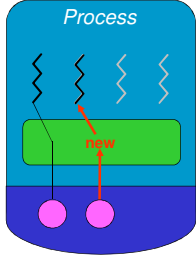

**Working principle**

- Blocking syscall scenario on 2 processors

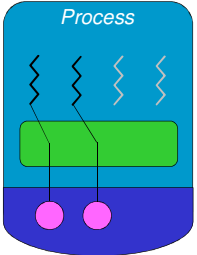

**Working principle**

- Blocking syscall scenario on 2 processors

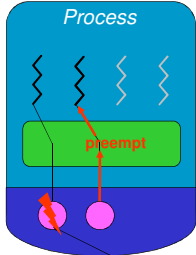
**Working principle**

- Blocking syscall scenario on 2 processors





**Working principle**

- Blocking syscall scenario on 2 processors



Preempt



Cse

### Working principle

- Blocking syscall scenario on 2 processors

The diagram shows a blue rounded rectangle labeled "Process". Inside, there are four vertical wavy lines representing instructions. A green horizontal bar is positioned below the instructions. A single pink circle representing a processor is at the bottom, connected to the green bar by a vertical line.

THE UNIVERSITY OF NEW SOUTH WALES

Cse

### Working principle

- Blocking syscall scenario on 2 processors

The diagram shows the same process as the previous slide. Now, two pink circles representing processors are at the bottom. The left one is connected to the green bar. The right one is also connected but has a vertical line extending downwards from it, labeled "Blocking syscall".

THE UNIVERSITY OF NEW SOUTH WALES

Cse

### Working principle

- Blocking syscall scenario on 2 processors

The diagram shows the process with two processors. The left processor is connected to the green bar. The right processor is also connected but has a vertical line extending downwards. A red arrow points to the green bar with the text "New + blocked".

THE UNIVERSITY OF NEW SOUTH WALES

Cse

### Working principle

- Blocking syscall scenario on 2 processors

The diagram shows the process with two processors. The left processor is connected to the green bar. The right processor is also connected but has a vertical line extending downwards. A red arrow points to the green bar with the text "I/O completion".

THE UNIVERSITY OF NEW SOUTH WALES

Cse

### Working principle

- Blocking syscall scenario on 2 processors

The diagram shows the process with two processors. The left processor is connected to the green bar. The right processor is also connected but has a vertical line extending downwards. A red arrow points to the green bar with the text "Unblocked".

THE UNIVERSITY OF NEW SOUTH WALES

Cse

### Working principle

- Blocking syscall scenario on 2 processors

The diagram shows the process with two processors. The left processor is connected to the green bar. The right processor is also connected but has a vertical line extending downwards. A red lightning bolt symbol is positioned between the two processors.

THE UNIVERSITY OF NEW SOUTH WALES

## Scheduler Activations

- Thread management at user-level
  - Fast
- Real thread parallelism via activations
  - Number of activations (virtual CPU) can equal CPUs
- Blocking (syscall or page fault) creates new activation
  - User-level scheduler can pick new runnable thread.
- Fewer stacks in kernel
  - Blocked activations + number of virtual CPUs

THE UNIVERSITY OF NEW SOUTH WALES

## Performance

Table IV. Thread Operation Latencies (µsec.)

Operation	FastThreads on Topaz Threads	FastThreads on Scheduler Activations	Topaz threads	Ultrix processes
Null Fork	34	37	948	11300
Signal-Wait	37	42	441	1840

THE UNIVERSITY OF NEW SOUTH WALES

## Adoption

- Adopters
  - BSD “Kernel Scheduled Entities”
    - Reverted back to kernel threads
  - Variants in Research OSs: K42, Barrelfish
  - Digital UNIX
  - Solaris
  - Mach
  - Windows 7
- Linux -> kernel threads

THE UNIVERSITY OF NEW SOUTH WALES

